La Contabilità col Personal Computer

STRUMENTI E TECNICHE DI PROGETTAZIONE PER PROGRAMMI GESTIONALI

Tullio Trincardi

GRUPPO EDITORIALE JACKSON

La Contabilità col Personal Computer

STRUMENTI E TECNICHE DI PROGETTAZIONE
DI PROGRAMMI GESTIONALI

di
Tullio Trincardi

GRUPPO
EDITORIALE
JACKSON

Via Rosellini, 12
20124 Milano

© Copyright per l'edizione originale: Gruppo Editoriale Jackson - Aprile 1985
COPERTINA: Silvana Corbelli
GRAFICA E IMPAGINAZIONE: Cristina De Venezia
COORDINAMENTO EDITORIALE: Daria Gianni
FOTOCOMPOSIZIONE: Lineacomp s.r.l. - Via Rosellini, 12 - 20124 Milano
STAMPA: Grafika '78 - Via Trieste, 20 - Pioltello (MI)

Tutti i diritti sono riservati. Stampato in Italia. Nessuna parte di questo libro può essere riprodotta, memorizzata in sistemi di archivio, o trasmessa in qualsiasi forma o mezzo, elettronico, meccanico, fotocopia, registrazione o altri senza la preventiva autorizzazione scritta dell'editore.

A Mariapia,
ad Annamaria

Ringrazio per l'aiuto prestato le signorine Enrica Menozzi e Laura Spaggiari della APPLE COMPUTER, la POOL INFORMATICA di Reggio Emilia per l'IBM e la SMC di Carpi per la TEXAS INSTRUMENTS.

Ringrazio in modo particolare la signora Patrizia Tommaselli Villa Aponte per la collaborazione prestata nella stesura di questo testo.

Tullio Trincardi.

SOMMARIO

PARTE I: Programmi generali per la gestione aziendale 1

CAPITOLO 1 **IL SISTEMA INFORMATIVO AZIENDALE** 3
1.1 Il sistema aziendale 3
1.2 Il sistema informativo 4
1.3 La integrazione del sistema informativo 6
1.4 La automatizzazione del sistema informativo 8

CAPITOLO 2 **LA PROGETTAZIONE DI UN PROGRAMMA** 9
2.1 I criteri di progettazione: funzionalità, modificabilità, usabilità 9
2.2 Le fasi della progettazione di programmi funzionali e modificabili 10
2.3 La strutturazione e la rappresentazione simbolica della struttura di programmi funzionali e modificabili 10

CAPITOLO 3 **LA PROGETTAZIONE DI PROGRAMMI INTERATTIVI** 15
3.1 Il colloquio macchina operatore realizzato mediante maschere di visualizzazione 15
3.2 La costruzione delle maschere 17

CAPITOLO 4 **IL CONTROLLO DELL'INPUT** 21
4.1 Il controllo dell'input dell'operatore 21
4.2 Il controllo ortografico 21
4.3 Il controllo logico 23

CAPITOLO 5 **LA ESECUZIONE DI MODULI DI PROGRAMMA GESTITA DAL MENU'** 31
5.1 La esecuzione di moduli di programma gestita da menù 31

CAPITOLO 6 **CARATTERI DEGLI ARCHIVI GENERALI NEL SISTEMA INFORMATIVO AZIENDALE** 37
6.1 Gli archivi generali e i caratteri di stabilità e accessibilità dei dati in essi archiviati 37
6.2 Il Profile: la sua funzione, la sua struttura 38
6.3 I programmi di gestione di un profile e degli altri archivi 40

CAPITOLO 7 **GLI ARCHIVI SPECIFICI** 45
7.1 I master file 45
7.2 I file transazionali 45
7.3 I file riassuntivi 45
7.4 I collegamenti tra i file 46
7.5 I modi di gestione dei file 47

CAPITOLO 8 **GLI STRUMENTI DI RAPPRESENTAZIONE DEL CONTENUTO DEGLI ARCHIVI** ... 49
8.1 I tracciati record ... 49
8.2 Le tavole delle variabili ... 49
8.3 Gli indici ... 50

CAPITOLO 9 **LISTATI DELLE SUBROUTINE DI BASE** ... 51
9.1 Organizzazione di un programma e distribuzione delle righe tra le sue fonti ... 51
9.2 La struttura e numeri di riga delle subroutine di base ... 51
9.3 I listati delle subroutine ... 52

CAPITOLO 10 **GLI OBBLIGHI CONTABILI** ... 59
10.1 I soggetti obbligati alla tenuta delle scritture contabili ... 59
10.2 I regimi di impresa ... 60
10.3 Gli obblighi di tenuta e conservazione delle scritture contabili ... 61

CAPITOLO 11 **L'AZIENDA COME OGGETTO DI OSSERVAZIONE E RILEVAZIONE** ... 63
11.1 L'azienda e le operazioni della sua gestione ... 63
11.2 La nozione di reddito globale e di reddito di esercizio ... 64
11.3 La composizionedel reddito di esercizio ... 65
11.4 Il metodo della partita doppia ... 66
11.5 La serie dei conti che consentono la duplice osservazione e rilevazione degli scambi ... 67
11.6 Il piano dei conti ... 69

CAPITOLO 12 **CONTABILITA' GENERALE: GLI OUTPUT RICHIESTI** ... 71
12.1 Il libro Giornale ... 71
12.2 I mastri di conto ... 72
12.3 I registri IVA ... 73
12.4 Sintesi dei dati richiesti in OUTPUT ... 74
12.5 Analisi dei dati richiesti in OUTPUT ... 75

CAPITOLO 13 **CONTABILITA' GENERALE: GLI INPUT** ... 77
13.1 La definizione dei file di input MOVIMENTI, CONTI, PROFILE ... 77
13.2 Le relazioni tra INPUT ed OUTPUT ... 77
13.3 Il file MOVIMENTI ... 78
13.4 Il file conti ... 81
13.5 Le relazioni tra file MOVIMENTI e file CONTI in un esempio di funzionamento ... 82

CAPITOLO 14 **IL PROGRAMMA DI APERTURA DEI CONTI** ... 85
14.1 Caratteristiche del programma ... 85
14.2 Caratteristiche dell'indice ... 85
14.3 Il programma ... 87

CAPITOLO 15 **LA REGISTRAZIONE DEI MOVIMENTI CONTABILI** 97
15.1 Oggetti di rilevazione di PRIMA NOTA 97
15.2 Uno schema di PRIMA NOTA 98
15.3 La registrazione dei movimenti contabili 99

CAPITOLO 16 **LE STAMPE DEI DOCUMENTI DI PROVA E DEI REGISTRI DELLE OPERAZIONI DI ESERCIZIO** 107
16.1 Schema generale dei diagrammi di struttura dell'output 107
16.2 Le subroutine di stampa 108
16.3 I programmi principali di visualizzazione o stampa 112

CAPITOLO 17 **LA CHIUSURA DEI CONTI E IL CALCOLO DEL REDDITO DI ESERCIZIO** 117
17.1 Le fasi di svolgimento delle operazioni di CHIUSURA 117
17.2 Il menù di CHIUSURA 117
17.3 La CHIUSURA dei conti 118
17.4 Il programma 119

PARTE II: APPENDICI 123

CAPITOLO 1 **NOZIONI FONDAMENTALI DI PROGRAMMAZIONE IN BASIC** 125
1.1 Il programma 125
1.2 La forma delle istruzioni di programma 126
1.3 I tipi di dati su cui un programma opera 127

CAPITOLO 2 **LE OPERAZIONI CON LE VARIABILI E LE COSTANTI** 131
2.1 Le operazioni sulle variabili numeriche 131
2.2 Le operazioni sulle variabili alfabetiche 133

CAPITOLO 3 **IL CONTROLLO DELLA SEQUENZA DELLE ISTRUZIONI DI PROGRAMMA** 137
3.1 Le istruzioni di modifica della sequenza di esecuzione di un programma 137
3.2 Le istruzioni di salto incondizionato 137
3.3 Le istruzioni di salto condizionato 140
3.4 Le istruzioni di Loop 140

CAPITOLO 4 **LE ISTRUZIONI DI TRASFERIMENTO E ASSEGNAZIONE** 145
4.1 Le istruzioni di assegnazione da programma LET, READ ... DATA, RESTORE, DIM, CLEAR 145
4.2 Le istruzioni di assegnazione da tastiera e da altra periferica INPUT e GET 147
4.3 Le istruzioni PRINT e le funzioni di tabulazione 148

CAPITOLO 5 **NOZIONI SUGLI ARCHIVI ELETTRONICI** 151
5.1 Gli archivi elettronici: la loro struttura e le modalità di accesso alle informazioni in essi contenute 151
5.2 I file sequenziali: la lettura e la registrazione 152
5.3 I file ad accesso casuale: la lettura e la registrazione 154

CAPITOLO 6 **LA STRUTTURA DEGLI ARCHIVI** 157
6.1 La rappresentazione della struttura degli archivi 157
6.2 Le tipiche strutture di organizzazione degli archivi 159

CAPITOLO 7 **I METODI DI ELABORAZIONE DELLE STRUTTURE SEQUENZIALI** 161
7.1 I programmi di ordinamento 161
7.2 I programmi di ricerca 164
7.3 L'inserimento e la cancellazione di un record 167

CAPITOLO 8 **METODI DI ELABORAZIONE DELLE STRUTTURE A LISTA E CASUALI** 169
8.1 La ricerca e la cancellazione di un record nelle strutture a lista 169
8.2 La determinazione di un indirizzo di un record mediante il metodo HASH-CODE 169

PARTE III: PROGRAMMI PER LA GESTIONE DI BANCHE DATI 171

CAPITOLO 1 **DAI PROGRAMMI IN BASIC A QUELLI IN DBASE** 173
1.1 Difficoltà di programmazione in BASIC o in altri linguaggi dello stesso tipo 173
1.2 Tipi e vantaggi dei DATA-BASE 173

CAPITOLO 2 **ELEMENTI GRAMMATICALI DI DBII** 175
2.1 I file 175
2.2 Le variabili di file (di campo) 177
2.3 Le variabili di memoria 177
2.4 Le costanti 177
2.5 I verbi di comando 177
2.6 Gli operatori 178
2.7 I pronomi ALL e NEXT < n > 181
2.8 Le preposizioni FOR < espressioni di relazione > 181
2.9 Simboli usati per indicare alcuni elementi grammaticali 182

CAPITOLO 3 **LA SINTASSI DI DBII** 183
3.1 La sintassi di DBII 183
3.2 Analisi delle proposizioni: le espressioni 183
3.3 Analisi delle preposizioni: le proposizioni di comando 184

CAPITOLO 4 **LA GESTIONE DEL FILE: I COMANDI FONDAMENTALI** 187
4.1 La creazione di un archivio 187
4.2 La visualizzazione della struttura 188
4.3 La copia 189
4.4 La modifica 189
4.5 La utilizzazione 189
4.6 La registrazione 190
4.7 La visualizzazione dei dati 190
4.8 L'ordinamento 191
4.9 La ricerca 192
4.10 La correzione 193

4.11 La cancellazione ... 194

CAPITOLO 5 **I COMANDI PER LA GESTIONE DEI FILE DI VARIABILI DI MEMORIA** ... 195
5.1 I comandi di inizializzazione e assegnazione ... 195
5.2 I comandi di salvataggio ... 196
5.3 I comandi di richiamo ... 196

CAPITOLO 6 **COMANDI DI INPUT-OUTPUT** ... 197
6.1 ACCEPT [" < messaggio > "] TO < VARIABILE > ... 197
6.2 INPUT [" < messaggio > "] TO < variabile > ... 197
6.3 WAIT TO < variabile > ... 197
6.4 SET FORMAT TO [SCREEN] [PRINT] [< format file >] ... 197
6.5 @ < n. riga, n. colonna > SAY "messaggio" ... 198
6.6 @ < n. riga, n. colonna > GET < variabile > ... 198
6.7 READ ... 198
6.8 Altri importanti comandi SET 198

CAPITOLO 7 **I PROGRAMMI: I FILE DI COMANDI** ... 199
7.1 I file di comandi ... 199
7.2 Come scrivere un file di comandi ... 199
7.3 La esecuzione dei file di comandi ... 200

CAPITOLO 8 **UN PROGRAMMA: STRUTTURA GENERALE** ... 203
8.1 I programmi: struttura per funzioni ... 203
8.2 Le funzioni generali ... 203
8.3 Le funzioni specifiche ... 204
8.4 La definizione delle fasi di realizzazione di un programma ... 205

CAPITOLO 9 **LA REALIZZAZIONE DEI SINGOLI MODULI DI PROGRAMMA** ... 207
9.1 La creazione della struttura dati ... 207
9.2 La codifica delle maschere video ... 208
9.3 La codifica del modulo di inizializzazione delle variabili ... 213
9.4 La codifica del modulo generale di controllo ... 214
9.5 La codifica dei moduli specifici ... 215
9.6 La codifica dei moduli che utilizzano più file: la registrazione dei movimenti di magazzino ... 219

PARTE IV: PROGRAMMI DI CALCOLO E SIMULAZIONE ... 225

CAPITOLO 1 **IL MULTIPLAN** ... 227
1.1 I fogli di lavoro elettronici ... 227
1.2 Il MULTIPLAN ... 228

CAPITOLO 2 **I COMANDI DEL MULTIPLAN** ... 231
2.1 I comandi: classificazione ... 231
2.2 Comandi di INPUT-OUTPUT ... 231

2.3 Comandi di formattazione dello schermo 233
2.4 Comandi di assegnazione e calcolo 234

CAPITOLO 3 **LE FORMULE** 237
3.1 Le formule 237
3.2 I modi di riferimento ai valori delle celle: il riferimento in modo assoluto e in modo relativo 237
3.3 Gli operatori aritmetici 238
3.4 Gli operatori logici 238
3.5 I principali operatori funzionali 238

CAPITOLO 4 **IL FOGLIO DI LAVORO ELETTRONICO** 241
4.1 Come si prepara un foglio elettronico 241
4.2 I moduli base 242
4.3 Moduli per la rappresentazione di serie temporali 242
4.4 Moduli contabili 244
4.5 La concatenazione di più moduli 248

CAPITOLO 5 **I PROBLEMI CARATTERISTICI** 251
5.1 Il calcolo dei giorni intercorrenti tra due date 251
5.2 La ricerca di un dato in una tavola di LOOKUP 254
5.3 L'ordinamento dei dati mediante la funzione SORT 257
5.4 La costruzione di modelli interattivi 262
5.5 Il trasferimento dei dati da DBASE al MULTIPLAN 265

Breve riepilogo dei comandi del Multiplan 271

INTRODUZIONE

Questo libro nasce dall'esigenza di dare sistematicità a quell'insieme di problemi che si presentano nel momento in cui si deve progettare una procedura gestionale automatizzata:

come analizzarne la struttura e le sue relazioni con il sistema di rilevazione di cui andrà a far parte?

quale ordine dare ai problemi da risolvere?

quali strumenti utilizzare nella loro soluzione?

Messi in rilievo i caratteri di sistematicità e le esigenze di integrazione delle procedure aziendali, ho cercato di illustrarne i fondamentali criteri di progettazione (la funzionalità, la modificabilità, la facilità di uso), le fasi, le tecniche di rappresentazione, le caratteristiche del colloquio macchina-operatore e le modalità di controllo della esecuzione dei programmi.

Ho quindi descritto le strutture dati più comunemente impiegate nelle nostre procedure con i loro vincoli ed i principali metodi di elaborazione.

Convinto che esistano in commercio programmi di grandi potenzialità e facilità d'uso per la soluzione di problemi gestionali specifici caratterizzati dalla prevalenza della gestione dati, o del calcolo e della simulazione, o della rappresentazione video-grafica, ho ritenuto di dare risalto, oltre che al BASIC, linguaggio tipico dei microcomputer, agli strumenti per la gestione di banche dati e di tabelle elettroniche.

Degli strumenti utilizzati sono descritte, oltre alle caratteristiche funzionali e alle concrete possibilità di applicazione, anche istruzioni e funzioni elementari; dei problemi economico-contabili sono illustrati i concetti fondamentali, non con l'intento di sostituire i tanti esaurienti manuali in circolazione, ma per dare, a coloro che avessero qualche dubbio sull'uno o l'altro argomento, la possibilità di trovare una spiegazione sufficientemente completa.

PARTE I

PROGRAMMI GENERALI PER LA GESTIONE AZIENDALE

CAPITOLO 1

IL SISTEMA INFORMATIVO AZIENDALE

1. 1 Il sistema aziendale
1. 2 Il sistema informativo
1. 3 La integrazione del sistema informativo
1. 4 La automatizzazione del sistema informativo.

1.1 Il sistema aziendale.

Si dice che la azienda per la sua attività, le persone che la realizzano, i beni di cui è composta, è un **sistema**.

Per sistema si intende un insieme di parti, legate da vincoli di complementarità e di connessione, volte ad un fine generale comune.

Un sistema è composto di parti con fini particolari, ma complementari rispetto al fine generale.

In un **sistema teorico** le parti mettono in relazione idee e concetti per spiegare e capire un fenomeno complesso.

In un **sistema pratico** le parti sono volte più o meno direttamente ad un fare.

La parte, il sottosistema, si denomina anche **processo**. Un processo ha input su cui esegue elaborazioni per produrre output; ogni fase ha feedback di controllo.

L'azienda persegue i suoi fini mediante l'attività ordinata e coerente di organi con funzioni particolari e complementari. Possiamo dire pertanto che l'azienda è un sistema, ma dobbiamo riconoscere che si tratta di un sistema complesso. Nell'azienda infatti si possono individuare diversi tipi di sistemi:
un sistema organizzativo, se la osserviamo negli organi che ne realizzano l'attività, **un sistema operativo** se la osserviamo nelle operazioni effettuate, **un sistema delle decisioni** se la consideriamo nelle decisioni prese per realizzare la attività aziendale. Infine, dato che le decisioni e le operazioni si realizzano sulla base di conoscenze, possiamo individuare **un sistema informativo** che dette conoscenze produce.

Possiamo immaginare che il sistema aziendale sia costituito di parti come si vede dalla figura 1.1 :

```
sistema  aziendale

                          |-------------------|
                          |   SISTEMA         |
 |-----------------------|   ORGANIZZATIVO   |----------------------|
 |                        |                   |                      |
 |                        |-------------------|                      |
 |                                                                   |
 |                                                                   |
 | |------------------|                        |------------------| |
 | |   SISTEMA        |                        |   SISTEMA        | |
 |-|   DELLE          |                        |   DELLE          |-|
 | |     DECISIONI    |                        |     OPERAZIONI   | |
 | |------------------|                        |------------------| |
 |                                                                   |
 |                                                                   |
 |                       |-------------------|                      |
 |                       |   SISTEMA         |                      |
 |----------------------|   INFORMATIVO     |----------------------|
                         |                   |
                         |-------------------|
```

Figura 1.1

In un sistema organizzativo sono strutturati gli organi dell'azienda, le persone che con diverse competenze e funzioni vi operano con vincoli di dipendenza e di complementarità e che ne guidano l'attività verso comuni fini aziendali.

In un sistema di decisioni si compongono, appunto, le decisioni per la realizzazione dei fini dei soggetti aziendali, perchè le singole decisioni sono o dovrebbero essere tra di loro coordinate e volte ad un fine comune.

Notiamo che decidere vuol dire scegliere per raggiungere degli obiettivi, perseguibili con diverse modalità di cui, in qualche modo, è possibile prevedere gli sviluppi futuri, coerentemente con dati principi di orientamento.

In un sistema operativo sono composte le operazioni aziendali, gli scambi, le operazioni di trasformazione interna, l'impiego dei fattori per la realizzazione dei processi e delle combinazioni produttive.

1.2 Il sistema informativo

I dati, espressione immediata dei fenomeni aziendali, i programmi di elaborazione, le informazioni prodotte, costituiscono il sistema informativo .

Il suo valore dipende dal valore delle informazioni prodotte e cioè dalle loro caratteristiche di tempestività, utilizzabilità, leggibilità, fondatezza.

In astratto si può dire che un sistema informativo prende dei dati in input, li elabora, crea degli output.

I dati in input possono essere dati numerici o alfabetici, comunicati da terminale o da tastiera, oppure dati analogici, trasmessi da strumenti di misura impiegati in processi industriali.

In un sistema aziendale i dati spesso originano da documenti quali gli ordini di acquisto, gli ordini di vendita, i cartellini di lavoro, le bolle di consegna, le fatture, le ricevute, etc.

L'elaborazione consiste prevalentemente di un lavoro di classificazione e di ordinamento, più raramente di calcolo.

L'output è dato dalle informazioni prodotte per il sistema organizzativo ,le decisioni che deve prendere e le operazioni che deve documentare.

Un sistema informativo non può però essere costituito soltanto delle funzioni di input, output ed elaborazione, mancano infatti alcune importanti funzioni di controllo di validità dei dati raccolti e delle informazioni prodotte.

Alcuni controlli sulla integrità e correttezza delle informazioni debbono essere interni agli stessi processi di elaborazione e di input: si parla di processi di feedback.

Altri controlli, questa volta esterni al sistema di elaborazione delle informazioni debbono pure essere predisposti per accertare regolarmente la validità delle informazioni stesse.

Un sistema informativo può allora essere rappresentato mediante la seguente figura.

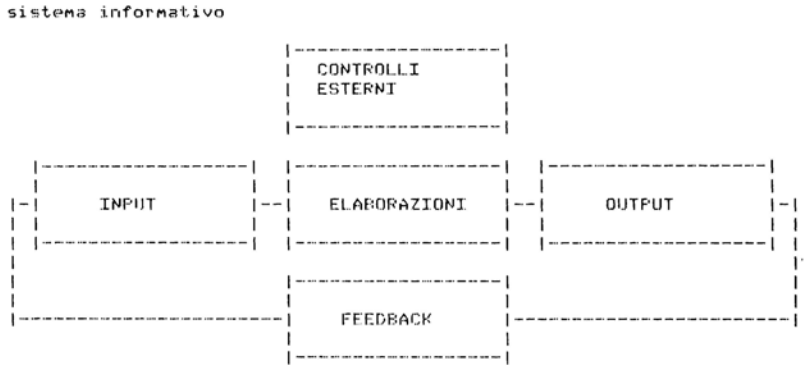


Figura 1.2

Un sistema informativo può essere considerato anche per le metodologie con cui vengono prodotte le informazioni. Allora nella unità del sistema il primo livello di osservazione consente di individuare quei sottosistemi che prendono il nome di procedure.

Con le procedure si perseguono fini particolari complessi, si pensi alla procedura PAGHE, CONTABILITA' GENERALE, CLIENTI, FORNITORI etc..

A sua volta una procedura è composta di programmi, di subroutine (procedura in senso stretto), di funzioni. Più programmi linkati costituiscono una procedura.

1.3 La integrazione del sistema informativo

Il sistema informativo aziendale ha il compito di provvedere alla raccolta delle informazioni e alla loro elaborazione per un efficiente controllo della gestione e una consapevole formazione delle decisioni. Il sistema è complesso, unitario il fine generale, parziali gli scopi delle sue parti.

Se in una azienda assumono peso rilevante i compiti di rilevazione contabile, altri compiti di controllo dei rapporti con la clientela e con i fornitori, di controllo dei magazzini e dei processi produttivi, di controllo e rilevazione del lavoro dei dipendenti, di analisi economiche e finanziarie, concorrono a definire gli scopi di un sistema informativo e dei suoi sottosistemi. Ciascuno di questi richiede input, provvede ad elaborazioni, produce informazioni e rapporti .

Spesso i dati prodotti in output da un sottosistema sono utilizzati come input dagli altri. Deve essere predisposto un collegamento, una interfaccia, tra il sottosistema e le altre parti con cui il sottosistema si trova collegato.

Se la comunicazione dei dati tra le diverse procedure avviene manualmente si dice che il sistema informativo è composto di **procedure separate**.

Se la comunicazione dei dati tra una procedura e l'altra avviene invece in modo automatico si dice che il sistema informativo è un **sistema integrato**.

Evidentemente la contabilità generale richiede dati da tutte le altre procedure, mentre a sua volta alle altre comunica dati. Esiste tra di esse un **sistema di interfacce**.
Infatti la contabilità generale produce gli estratti conto, le situazioni contabili, i prospetti dei conti Profitti e Perdite e di Stato Patrimoniale e altri dati che trasmette in particolare alle procedure di analisi economica e finanziaria per la formazione delle decisioni aziendali. Richiede dati riassuntivi dalle procedure di contabilità clienti, da quelle di magazzino, di paghe e stipendi. Così il controllo clienti fonda il suo funzionamento sui dati raccolti dagli ordini dei clienti, sulle fatture emesse, sui documenti di incasso che gli sono propri, ma deve ricevere dei dati sulle consegne di merci ordinate dalle procedure di controllo del magazzino.

Similmente operano le procedure di controllo dei fornitori.

La procedura di controllo di magazzino solitamente non si occupa soltanto delle rilevazioni di inventario, per la individuazione dei collocamenti della merce negli stocks, per la determinazione dei movimenti di merce e della loro disponibilità.

A volte deve dare indicazioni sui livelli di riordino e sui lotti più economici di acquisto, e sempre deve dare avviso della avvenuta esecuzione delle consegne.

Evidentemente una procedura di magazzino deve essere interfacciata direttamente con le procedure clienti e fornitori e, in misura più limitata, con la contabilità generale per la comunicazione, tra l'altro, degli scarti, sfridi etc.

Spesso un altro sottosistema inportante è quello rivolto alla analisi finanziaria ed economica della attività aziendale, volto a consentire la formazione delle decisioni con applicazioni specifiche che riguardano il controllo dei costi e dei ricavi, le previsioni e i budget aziendali.

Evidenti sono le necessità di interfaccia di questo particolare sottosistema con le altre procedure.

Il sistema, nel suo insieme e nelle sue relazioni può essere rappresentato ricorrendo al diagramma seguente in cui i sottosistemi sono rappresentati da rettangoli, le applicazioni particolari da ovali, le relazioni tra le parti da linee continue:

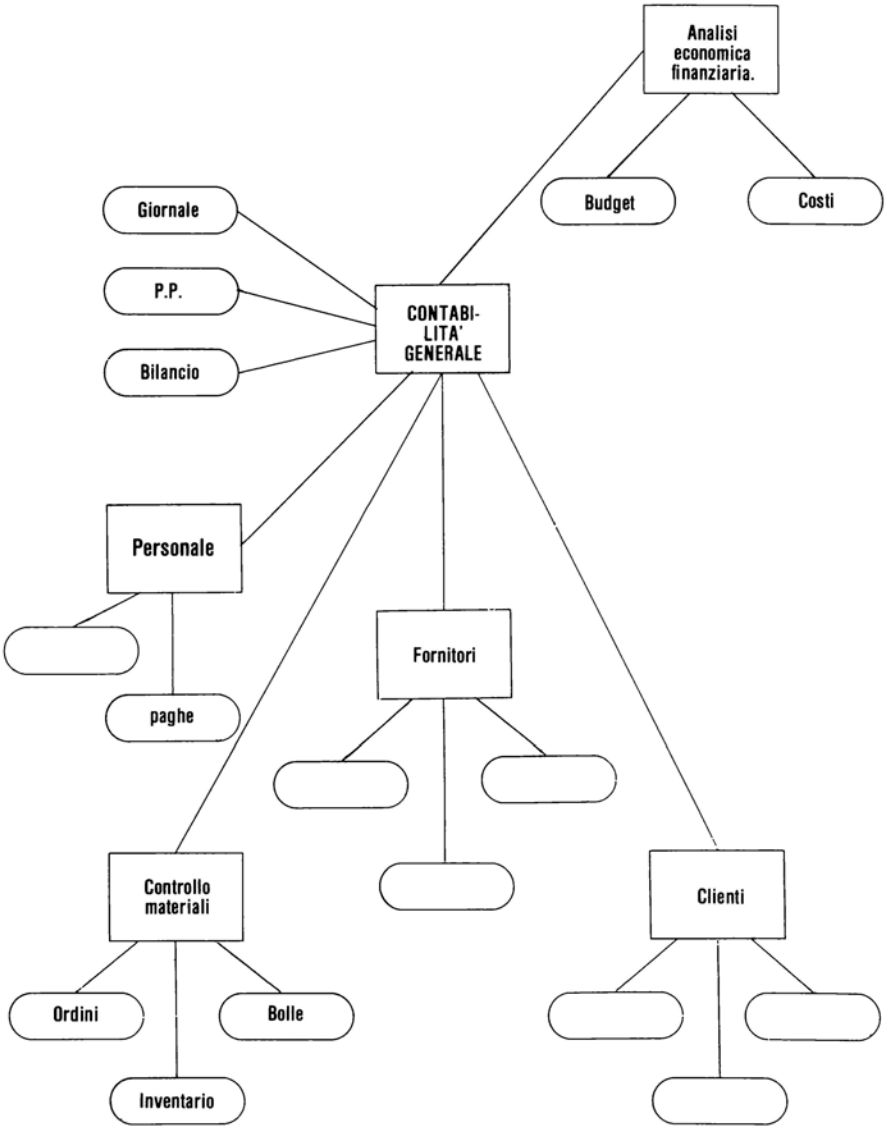


Figura 1.3

La creazione di un sistema informativo, o di alcune procedure richiede la conoscenza del sistema aziendale in particolare per la parte dei problemi che si debbono risolvere. Richiede **l'analisi del sistema**.

L'analisi consiste nella scomposizione della struttura aziendale nella definizione delle funzioni svolte dalle parti, individuando quei programmi e quelle procedure che possono soddisfarne le esigenze informative.

Più in particolare si può dire che è necessario **analizzare il sistema delle decisioni** i cui problemi informativi si debbono risolvere, **definire il flusso delle informazioni** che potrebbero soddisfare tali esigenze, **definire i dati necessari** a produrre tali flussi informativi, **determinare i costi del sistema** e della parte creata.

Nel suo insieme il sistema deve risolvere i problemi informativi della azienda, fornendo con tempestività ed efficienza le informazioni necessarie alla formazione delle decisioni e alla esecuzione delle operazioni aziendali.

Le qualità che fanno efficiente un sistema e le sue parti debbono evidentemente essere tenute in considerazione nei momenti in cui il sistema o anche le singole procedure vengono progettate.

In sintesi si dovrebbe tener conto del fatto che la qualità delle procedure prodotte è valutabile sulla base dei costi sostenuti e dei prodotti ottenuti, per cui hanno peso tutte quelle caratteristiche che fanno i programmi più economici, più affidabili e facili da usare.

Ha quindi valore particolare la **facilità d'uso**, la **manutenibilità**, la **portabilità**, la **modificabilità** delle procedure progettate.

1.4 L'automatizzazione del sistema informativo.

L'automatizzazione del sistema informativo si serve di elaboratori e può riguardare sia l'elaborazione, la comunicazione e la stampa delle informazioni, che l'acquisizione e la registrazione dei dati. In modo automatizzato possono essere emessi i documenti di prova delle operazioni compiute, registrati ed elaborati i dati relativi.

Solitamente la automatizzazione di un sistema informativo procede per gradi e non con continuità.

I manager spesso sentono la necessità di risolvere alcuni problemi informativi particolari e ne chiedono la automatizzazione. Coprono aree sempre più vaste del sistema informativo.

Presto nascono esigenze di integrazione e se le singole parti del sistema informativo non sono state progettate tenendo conto delle loro relazioni, della unità cui appartengono, del loro e del suo sviluppo, il sistema informativo deve essere ricreato tenendo conto delle nuove esigenze di integrazione e funzionalità. Necessariamente lo sviluppo del sistema informativo concorre a modificare la stessa struttura organizzativa della azienda.

In genere viene richiesta in primo luogo la automatizzazione delle paghe, poi vengono affrontati i problemi del magazzino, della fatturazione del controllo clienti e fornitori. Si deve tener conto del fatto che la procedura di magazzino non è volta soltanto alla registrazione dei movimenti, ma deve dare indicazioni di convenienza così come la procedura clienti e quella fornitori debbono, oltre alle più usuali situazione della clientela e dei fornitori, dare evidenza ai dati che possono suggerire il modo più conveniente di operare con loro.

Si ripete: se la automatizzazione non è pensata con riferimento alla unità del sistema e al suo sviluppo si raggiunge presto il momento in cui tutte le sue parti debbono essere ricostruite e integrate.

CAPITOLO 2

LA PROGETTAZIONE DI UN PROGRAMMA

2.1. I criteri di progettazione: funzionalità, modificabilità, usabilità.
2.2. Le fasi della progettazione di programmi funzionali e modificabili.
2.3. La strutturazione e la rappresentazione simbolica della struttura di programmi funzionali e modificabili.

2.1.Criteri di progettazione: funzionalità, modificabilità, usabilità

Le qualità che fanno il valore di un sistema informativo danno valore anche al singolo programma.
Ogni programma pertanto deve avere le qualità seguenti:

deve essere funzionale;
deve essere adattabile e modificabile;
deve essere facile da usare;
deve essere economico nell'acquisto e nella manutenibilità.

Un programma è funzionale se è in grado di soddisfare le esigenze dirette per cui può venire impiegato, senza errori, interruzioni per manovre sbagliate o altro.
Almeno per quanto riguarda l'accesso agli archivi generali la funzionalità implica che vi sia compatibilità con gli altri programmi del sistema.

La adattabilità e modificabilità è richiesta dal modificarsi delle situazioni in cui il sistema opera. Diventa facile se è prevista da programma la modifica di alcuni paramentri da parte dell'operatore (si pensi alla modifica delle aliquote I.V.A.o degli scaglioni dei redditi I.R.P.E.F.). Può essere possibile la stessa modifica del programma se questo è costruito per moduli ben documentati e non come un unico insieme.

La facilità d'uso dovrebbe essere un carattere della stessa funzionalità. Nel concreto si esplica attraverso la semplicità del colloquio tra elaboratore ed operatore utente, la semplificazione e riduzione al minimo delle operazioni richieste allo stesso, la possibilità di guidare, almeno in parte, la successione delle operazioni da eseguire.

2.2.Le fasi della progettazione di programmi funzionali e modificabili.

Un programma è uno strumento di soluzione di un problema e prima di essere definito richiede una chiara definizione e rappresentazione dello **scopo** da raggiungere.

Detto scopo può essere desunto dagli **output** richiesti dal proponente.

Lo scopo proposto è solitamente complesso e composto di scopi più semplici, il problema proposto non è un problema, ma un problema costituito di tanti problemi più semplici, composti di problemi elementari.

La stesura di un progetto di soluzione richiede quindi in primo luogo la **definizione dello scopo generale** da perseguire, quindi la sua **scomposizione in scopi più semplici**. E' necessaria una strutturazione del problema e probabilmente una rappresentazione simbolica della struttura.

Definiti gli scopi devono **essere definiti i dati, gli input** di cui si dispone. A questo punto è possibile **definire quali operazioni si debbono fare** per risolvere il problema proposto nelle sue parti e nell'insieme.

Possiamo dire quindi che l'analisi di un problema deve essere affrontato seguendo le seguenti fasi:

1) descrizione dell'output richiesto (costruzione di un diagramma di struttura dell'output);

2) descrizione dell'input disponibile;

3) determinazione della corrispondenza tra input ed output (costruzione del diagramma di struttura finale);

4) determinazione delle azioni da far eseguire al programma;

5) pseudocodifica e codifica con definizione dei nomi e dei tipi di variabili da usare.

2.3.La strutturazione e la rappresentazione simbolica della struttura di programmi funzionali e modificabili.

Si debba preparare un programma per la soluzione di un problema; non un programma qualsiasi che funzioni e risponda solo alle esigenze esplicitamente dichiarate dal problema stesso, ma un programma corretto, il che è diverso e più completo.

È importante riconoscere la differenza che esiste tra un programma che funziona e un programma corretto. Se un programma che non funziona è indubbiamente un cattivo programma, non è detto che un programma che funzioni sia, al contrario, un programma corretto. In effetti il programma può presentare i seguenti difetti:

può essere difficile da capire
può essere difficile da mantenere
può essere difficile da modificare.

Tali difetti sono difetti gravi che è necessario evitare proprio per non compromettere la funzionalità dei programmi e ciò è possibile se si seguono le norme di progettazione dettate dalle tecniche di programmazione strutturata.

Le idee su cui si fonda nascono dal fatto che i problemi sono scomponibili in parti e che i vincoli di struttura tra le parti possono essere rappresentati ricorrendo a tre soli tipi di vincolo: la **sequenza**, la **ripetizione** e la **selezione**.

La strutturazione di un programma richiede si proceda nel modo seguente:

1) Ciascun problema deve essere scomposto in strutture gerarchiche di parti, così come il relativo programma;

2) ogni parte o blocco logico deve essere legata ai blocchi di scomposizione da vincoli dello stesso tipo;

3) i vincoli tra blocco superiore e blocchi di scomposizione debbono essere limitati ai tipi sequenza, ripetizione, selezione;

4) si debbono evitare il più possibile le istruzioni di GO TO.

La rappresentazione del problema generale e dei blocchi di scomposizione viene effettuata mediante rettangoli:

Figura 2.1

Nella rappresentazione di un problema si parte dall'alto con il primo blocco al centro indicante il problema generale.

Si passa al primo livello di scomposizione del problema in parti indicandone la successione logica mediante la distribuzione dei blocchi da sinistra a destra. I vincoli di sequenza tra blocco superiore e blocchi di scomposizione vengono indicati mediante linee continue:

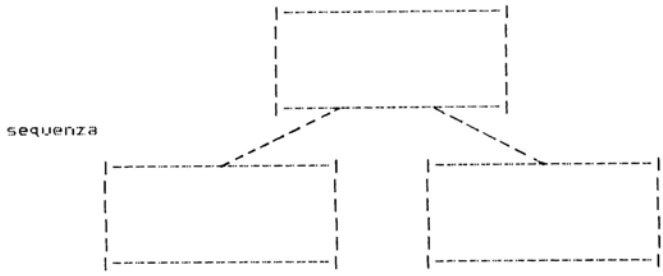


Figura 2.2

La ripetizione viene indicata unendo il blocco superiore a quelli o a quello di ripetizione con una linea ed un asterisco:

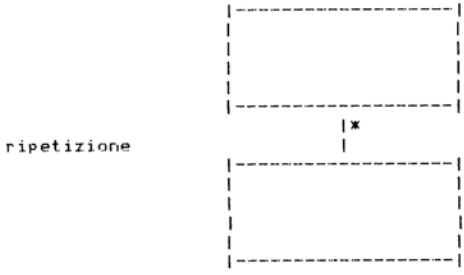


Figura 2.3

La selezione viene indicata unendo il blocco superiore a quelli inferiori, con cui è in vincolo di selezione, mediante una linea ed una "o":

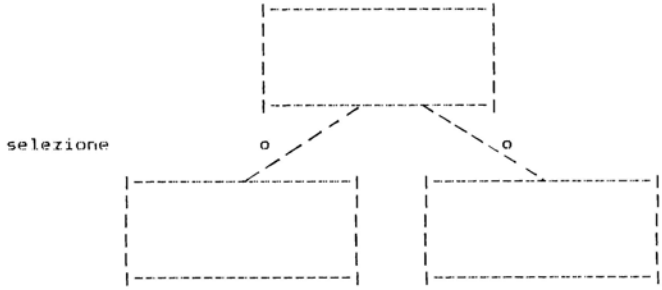


Figura 2.4

Si ricordi che ciascun blocco superiore può essere legato a quelli inferiori che ne sono la scomposizione solo da una delle tre forme di struttura citate.
Si fa osservare inoltre che talvolta è necessario inserire nella struttura blocchi fittizi per evitare che i blocchi superiori siano legati a quelli inferiori da legami non omogenei.

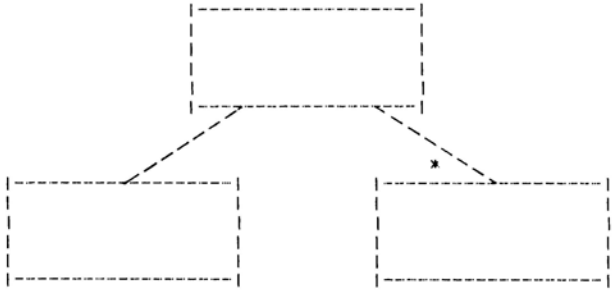


Figura 2.5

Si deve allora impiegare il diagramma di cui alla figura 2.6

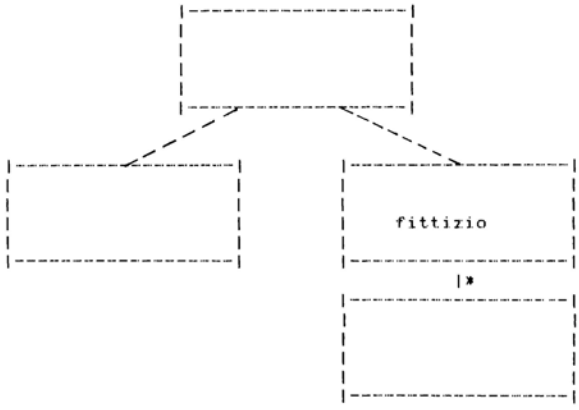


Figura 2.6

CAPITOLO 3

LA PROGETTAZIONE DI PROGRAMMI INTERATTIVI

3.1 Il colloquio macchina operatore realizzato mediante maschere di visualizzazione
3.2 La costruzione delle maschere:
3.2.1 la scelta e assegnazione delle scritte;
3.2.2 la tabulazione delle scritte;
3.2.3 la predisposizione dei controlli di input;
3.2.4 la visualizzazione delle scritte.

3.1.Il colloquio macchina operatore realizzato mediante maschere di visualizzazione.

Un programma che consente un colloquio con l'operatore, gli espone le necessità del momento e ne ascolta le richieste, è solitamente un programma più facile da usare di quei programmi in cui il colloquio non esiste e la successione e i tempi delle operazioni sono definiti rigidamente.

Il colloquio tra operatore e computer si svolge, in genere, mediante un video che visualizza i messaggi dell'elaboratore e la tastiera che raccoglie le risposte dell'operatore.

I messaggi non vengono di solito visualizzati utilizzando in sequenza la successione degli spazi liberi dello schermo, ma predisponendolo appositamente per il tipo di colloquio cui deve servire. Si parla di format-video o di maschere di visualizzazione.

Essendo diversi gli oggetti dei colloqui possiamo distinguere tra diversi tipi di maschere, ad esempio per la **consultazione** degli archivi, per i **menù di scelta** delle opzioni di esecuzione, per **la raccolta dati**, per lo **scambio interattivo di informazioni**.

Programmi predisposti per consentire simili tipi di colloquio si dicono interattivi perchè consentono una interazione tra macchina e operatore.

E' evidentemente necessario per tutti i tipi di colloquio, ma in particolare in quelli per la raccolta dati, che da un lato il programma definisca con chiarezza i dati che richiede, dall'altro che provveda al loro controllo per impedire vengano immessi dati non coerenti con quelli richiesti. La richiesta di dati avviene solitamente attraverso video-terminali, mentre l'immissione viene effettuata di solito da tastiera e da dischi, o, via cavo, da altri terminali. E' pertanto sul video che i dati richiesti debbono essere definiti ed all'atto del loro ricevimento in memoria che possono essere controllati.

La definizione delle caratteristiche dei dati in input avviene allora mediante maschere di input in cui lo schermo video viene utilizzato per la formulazione delle domande e la visualizzazione delle risposte.

Il controllo dei dati immessi viene effettuato da programma con apposite routine di controllo dei dati ricevuti in memoria. Detto controllo riguarda in primo luogo la loro **ortografia**, in secondo la loro **coerenza e logicità**.

Il controllo ortografico viene in genere rivolto ad impedire la immissione di dati numerici al posto di dati alfabetici e viceversa.

Il controllo di coerenza riguarda in genere la correttezza delle date immesse delle partite I.V.A., del codice fiscale etc..

Un menù è costituito di solito da una successione di scritte indicanti opzioni di esecuzione e di una riga per la risposta dell'operazione.

```
-------------------------------------------------
|                                               |
|                                               |
|                                               |
|                                               |
|    1 - Bolle                                  |
|                                               |
|                                               |
|    2 - Fatture                                |
|                                               |
|                                               |
|    3 - Fine lavoro                            |
|                                               |
|                    Dammi la tua opzione       |
|                                               |
|_______________________________________________|
```

Figura 3.1

Una maschera di input riguarda di solito un insieme omogeneo e limitato di operazioni. E' costituita di una serie di brevissime scritte di richiesta di input ordinatamente disposte sullo schermo, in una successione spaziale che corrisponde, di solito, alle operazioni da fare. Talvolta un cursore lampeggiante indica, nell'insieme delle operazioni richieste, quella che nel momento deve essere effettuata.

Le risposte dell'operatore vengono visualizzate accanto alle scritte di richiesta dati.

```
LISTA FORNITORI

CODICE   101/3

DESCRIZIONE  Maxibox  da 30 in PVC con chiusu
             ra automatica 30*30*20

FORNITORI
    ICAM S.P.A VIA BONOMI 3 PAVIA TEL     35809
    REGGEM S.P.A. VIA RANDI 25 REGGIOLO  36045
    RUBBER S.P.A. VIA SOLLI 54 GAUSTALLA 34580
```

Figura 3.2

Di solito case di software forniscono appositi sofisticati programmi per la costruzione di maschere.Talvolta si deve provvedere direttamente a preparare apposite routine per la loro costruzione e visualizzazione.

3.2 La costruzione delle maschere

Individuato un insieme omogeneo e limitato di operazioni da svolgere in modo interattivo la costruzione delle maschere richiede:

a) che venga definito quali messaggi è utile far apparire per ogni videata;

b) che venga definita la posizione in cui ciascuna scritta deve apparire;

c) che siano approntate le necessarie routine di controllo ortografico;

d) che siano approntate le necessarie routine di controllo logico.

e) che siano approntate le routine di visualizzazione delle scritte di maschera e degli input dell'operatore.

3.2.1 La scelta e assegnazione delle scritte.

Individuato un insieme omogeneo e limitato di operazioni da far svolgere all'operatore devono essere visualizzate le informazioni essenziali e necessarie alla loro esecuzione. Dette informazioni costituiscono dei messaggi per l'operatore che debbono

essere concisi, di immediata utilizzazione e inequivoci. Detti messaggi possono essere assegnati a delle variabili da visualizzare. La assegnazione può essere effettuata in BASIC mediante le istruzioni READ.... DATA, quando sia definito con precisione il numero delle scritte da assegnare e il loro contenuto.

Si utilizzi per definire il numero delle scritte della maschera la variabile NINPUT% e per la assegnazione delle scritte la variabile multipla TMS$().

Se la subroutine inizia da riga 16500, questa può iniziare con le istruzioni seguenti:

```
16500 REM MASCHERA N.1 MENU'
16510 FOR I=1 TO NINPUT%:READ TMS$(I):NEXT I
16520 DATA " TITOLO MENU","SECONDA SCRITTA","TERZA SCRITTA",
"QUARTA SCRITTA CON RICHIESTA DI OPZIONE"
```

3.2.2 La tabulazione delle scritte.

Si deve ora decidere dove far apparire le scritte. Esistono norme pratiche di distribuzione delle scritte che è opportuno seguire per rendere efficienti i format-video.

Ad esempio si dice che è bene fissare in alto a sinistra il punto di partenza della maschera. Che è opportuno realizzare un equilibrio simmetrico tra le parti dello schermo, centrando i titoli in alto e distribuendo le informazioni similari sui due lati dell'asse verticale centrale; inoltre che è preferibile dare una informazione su una sola linea senza spezzarla in due linee successive.

I valori di tabulazione orizzontali e verticali, determinati eventualmente con appositi prospetti quadrettati (vedasi fig.3.3), debbono essere assegnati ad apposite variabili multiple che nel nostro caso definiamo con V%() e H%().

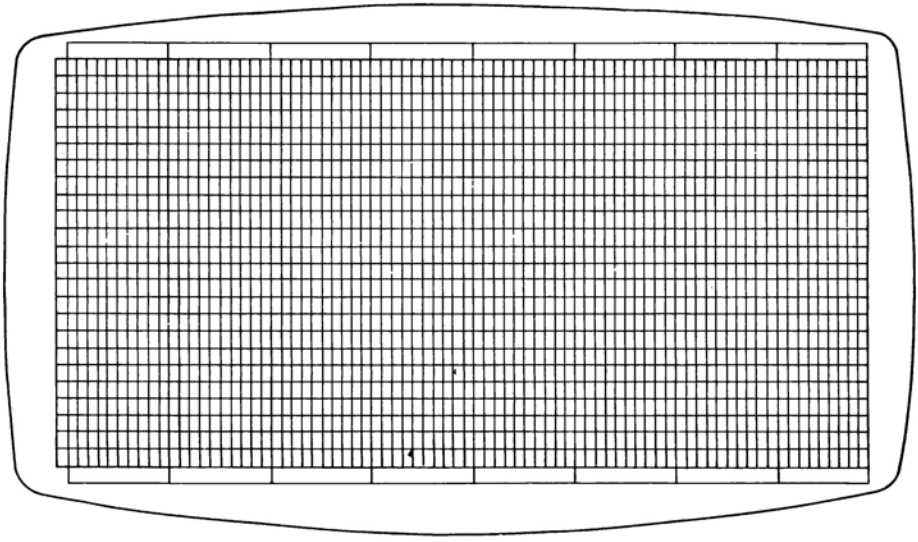


Figura 3.3

Ancora possiamo utilizzare le istruzioni READ... DATA.
Avremo le seguenti righe :

```
16530 FOR I= 1 TO NINPUT%:READ V%(I):NEXT I:DATA 4,8,12,16
16540 FOR I= 1 TO NINPUT%:READ H%(I):NEXT I:DATA 10,7,7,7
```

Il contenuto e la funzione delle variabili necessarie per la costruzione di una maschera viene in sintesi descritta dalla seguente figura:

```
Vettore V% ( ) per i valori                    Vettore TMS$( ) per i
di tabulazione verticale                       messaggi da programma
                 ------------------------------
 V%(1)=3         |                            | TMS$(1)="1 - Bolle"
 V%(2)=6         |                            | TMS$(2)="2 - Fatture"
 V%(3)=9         |   1 - Bolle                | TMS$(3)="3 - Fine lavoro"
 V%(4)=11        |                            | TMS$(4)=" Dammi la tua op
                 |                            |
                 |   2 - Fatture              |
                 |                            |
                 |                            |
                 |   3 - Fine lavoro          |
                 |                            |
                 |       Dammi la tua opzione |
                 |____________________________|

Vettore H%( )per i valori
di tabulazione  orizzontale :  H%(1)=4:H%(2)=4:H%(3)=4:H%(4)=8

Vettore TIPO% ( ) per il tipo di    Vettore NCAR%( ) per il numero massimo di
caratteri accettabili in input:     caratteri accettati in risposta:


     TIPO%(1)=0                            NCAR%(1)=0
     TIPO%(2)=0                            NCAR%(2)=0
     TIPO%(3)=0                            NCAR%(3)=0
     TIPO%(4)=1                            NCAR%(4)=1
```

Figura 3.4

3.2.3 La predisposizione dei controlli.

La predisposizione dei controlli di input viene effettuata definendo i tipi e il numero dei caratteri che verranno accettati in input dalle apposite routine di controllo.
Si utilizzino le variabili TIPO%() per la assegnazione del tipo numerico o alfabetico dei caratteri da immettere e le variabili NCAR%() per la definizione del numero dei caratteri accettati in ciascuna risposta dell'operatore.
Si assegni quindi alla variabile TIPO%(2) il valore 0, se il messaggio 2 vuole una risposta alfabetica, il valore 1, se vuole una risposta numerica.
Si hanno allora le seguenti righe:

```
16550 FOR I= 1 TO NINPUT%:READ TIPO%(I):NEXT I:DATA 0,0,0,1
16560 FOR I= 1 TO NINPUT%:READ NCAR%(I):NEXT I:DATA 0,0,0,2
```

3.2.4 La visualizzazione delle scritte.

Per la visualizzazione dei messaggi del calcolatore si ricorre ad una apposita subroutine. Detta subroutine deve provvedere a pulire lo schermo dalle scritte precedenti, a visualizzare i nuovi messaggi nelle posizioni definite dai valori di tabulazione di ciascuno di essi, a calcolare la posizione di visualizzazione dei messaggi di risposta. Per la cancellazione dello schermo in Basic MS DOS si utilizza la istruzione CLS, in Applesoft si utilizza la istruzione HOME.
Per la tabulazione in Basic MS DOS si impiega la istruzione LOCATE, con i parametri di tabulazione verticale e orizzontale e un terzo valore per la visualizzaione del cursore. Detti parametri sono nel nostro caso assegnati rispettivamente alle variabili V%() e H%().

Alla istruzione LOCATE,,, corrispondono in Basic Applesoft le istruzioni VTAB() e HTAB(). Nel nostro caso avremo VTAB(V%()) e HTAB(H%()).
Si deve ricordare che alcuni tipi di APPLE hanno uno schermo standard di 24 righe per 40 colonne.

Il calcolo delle posizioni dei messaggi di risposta deve essere fatto calcolando la lunghezza dei messaggi del calcolatore per consentire di collocare vicino ad essi le risposte dell'operatore. Per detto calcolo si impiega la istruzione LEN applicata alle variabili cui è assegnato il messaggio.

Le istruzioni di richiamo della subroutine di visualizzaione e di conclusione della subroutine di costruzione della maschera sono:

```
16570 GOSUB 11000'VISUALIZZAZIONE
16580 RETURN
```

Le istruzioni di visualizzazione sono:

```
11000 REM VISUALIZZAZIONE MASCHERE
11005 REM IN INPUT V%(),H%(),TMS$().IN OUTPUT NUOVA POSIZIONE ORIZ-
ZONTALE.
11010 CLS' PER PULIRE LO SCHERMO
11020 FOR I=1 TO NINPUT%
11030 LOCATE V%(I),H%(I),0:PRINT TMS$(I)
11040 H%(I)=H%(I)+LEN(TMS$(I))+2
11050 NEXT I
11060 RETURN
```

CAPITOLO 4

IL CONTROLLO DELL'INPUT

4.1 Il controllo dell'input dell'operatore.
4.2.Il controllo ortografico.
4.3 Il controllo logico:
4.3.1 il controllo della data.
4.3.2 il controllo della partita I.V.A.
4.3.3 il controllo del codice fiscale.,

4.1 Il controllo dell'input dell'operatore.

Sugli input dell'operatore è possibile effettuare sia un controllo ortografico che un controllo di coerenza e logicità. Sempre deve essere effettuato il controllo ortografico, meno frequentemente viene effettuato quello di coerenza.

Il controllo ortografico viene effettuato automaticamente sui singoli caratteri immessi dall'operatore, il controllo logico della risposta riguarda la coerenza con la domanda posta dall'elaboratore e con le altre risposte precedentemente date; talvolta viene fatto automaticamente, talaltra viene lasciato alla attenzione dell'operatore.

4.2 Il controllo ortografico.

Il controllo ortografico riguarda la corrispondenza tra tipi di caratteri immessi e tipi di caratteri richiesti, tra numero massimo di caratteri richiesti e numero massimo di caratteri immessi.

Il controllo ortografico si realizza, di solito, nelle fasi seguenti:

a) controllo di corrispondenza tra tipo di caratteri richiesti e tipo di caratteri immessi. Viene effettuato carattere per carattere, ove manchi la necessaria corrispondenza, l'immissione viene respinta e viene richiesta l'immissione di un carattere corretto;

b) visualizzazione carattere per carattere dei dati immessi e controllo del numero dei caratteri di input in modo che il loro numero corrisponda a quello richiesto;

c) accettazione o rifiuto dei dati immessi con correzione

a) Controllo del Tipo

Sia definito il tipo dei caratteri che vogliamo immettere, assegnando il valore 0 o 1 ad una variabile (nel nostro caso TIPO%()). Si decida per la immissione di caratteri numerici assegnando a TIPO%() il valore 1.

Si prenda in input un carattere alla volta mediante la istruzione INPUT$(1), o INKEY$, o, per l'Applesoft, GET.

Si assegni il carattere in input ad una variabile di controllo. Nel nostro caso prendiamo IN$(4).

Si confronti il contenuto della variabile con l'insieme dei caratteri ASCII che ci interessano e cioè quelli minori di "9" e maggiori di "0", e se il carattere appartiene al nostro insieme lo si sommi al contenuto di una variabile di accumulo dei caratteri richiesti dalla nostra risposta. Se invece il nostro carattere non appartiene all'insieme desiderato lo si respinga e si chieda un nuovo carattere.

Le istruzioni fondamentali sono le seguenti:

```
11250.......:IN$(4)=INPUT$(1)
11260
11270....
11280 IF TIPO%(C)=1 AND (IN$(4)<"0")OR(IN$(4)>"9")THEN 11250
 11290 IN$(5) = IN$ (5) + IN$ (4)
```

b) Visualizzazione carattere per carattere dei dati immessi e controllo del loro numero massimo

Si assegni ad una variabile (NCAR%(C)) il numero massimo dei caratteri da accettare in input per la risposta C.

Si predisponga un ciclo di richiamo dei caratteri di input con una variabile di controllo (J) che vari da uno al numero massimo dei caratteri da immettere.

Sia definito nel tredicesimo carattere ASCII il carattere che segnala alla macchina la fine di un input.

Sia definito nell'ottavo carattere ASCII il carattere che segnala alla macchina l'intenzione di riimmettere un carattere errato.

Se il carattere da visualizzare è in IN$(4) la sua posizione orizzontale nello schermo è pari al valore di tabulazione orizzontale (H%()) più il valore della variabile di controllo del ciclo di immissione (J).

Se il carattere di risposta è CHR$(13) la immissione è finita e si deve restituire il controllo al programma che ha richiesto l'input per la sua continuazione.

Se il carattere di risposta in IN$(4) è uguale a CHR$(8) allora si deve cancellare il carattere precedentemente visualizzato e visualizzare un nuovo carattere di input. Questa operazione richiede che il valore della variabile di controllo del numero dei

caratteri finora immessi sia diminuito di 2 (uno per il carattere errato più uno per il segnale di bakspace) e che della variabile di accumulo dei caratteri di risposta (IN$(5) siano presi solo i caratteri validi.

Le istruzioni fondamentali potranno essere le seguenti:

```
11200 REM INPUT SU VIDEO
11210
11220 IN$(5)="":IN$(4)="":J1=1
11230 FOR J=J1 TO NCAR%(C)
11240
11250 .....:IN$(4)=INPUT$(1)
11260 IF IN$(4)=CHR$(13) THEN RETURN
11270 IF IN$(4)=CHR$(8) AND J>=2 THEN J=J-2:IN$(5)=LEFT $(IN$(5),J)....
11275....
11280....
11285 IN$(5)=IN$(5)+IN$(4)
11290 LOCATE V%(C),H%(C),1:PRINT IN$(4)
11300 NEXT J:BEEP
11310
11350 RETURN
```

Nell'esempio completo si farà vedere più in dettaglio anche la tecnica di visualizzazione e cancellazione dei caratteri numerici che, per ragioni di allineamento, debbono essere visualizzati da destra a sinistra e non da sinistra a destra.

c) Accettazione o rifiuto dell'operatore dell'insieme dei dati immessi.

Alla fine dell'input può essere consentita all'operatore una verifica sulla validità dell'insieme dei caratteri immessi per una data risposta.

Le istruzioni per la richiesta di accettazione sono le seguenti:

```
11310 LOCATE 23,67,1:PRINT"O (Y/N)";:IN$(4)=INPUT$(1):IF IN$(4)="N" THEN
11210
```

Infine si restituisce il controllo al programma chiamante con la istruzione RETURN.

4.3 Il controllo logico:

4.3.1 Il controllo della data.

Un controllo logico si realizza mediante definizione a priori delle possibili risposte e accertamento della appartenenza della risposta data all'insieme previsto. Oltre ad una coerenza interna di ciascuna risposta si può talvolta accertare anche la coerenza e

logicità di due o più risposte tra di loro. Un caso comune è dato dai controlli effettuati sulle date immesse dall'operatore in cui la risposta alla domanda: "anno?" deve essere data, all'interno di un insieme definito, la risposta alla domanda: "mese?" non può essere superiore a 12 o minore di 1, la risposta alla domanda: "giorni?" non può essere inferiore ad 1 e superiore al numero dei giorni del mese, definito mediante la risposta precedente. Inoltre si fa osservare che il numero dei giorni di febbraio dipende dall'essere l'anno bisestile o no.

Si debbono quindi predisporre dei controlli interni alle singole risposte e dei controlli collegati alle risposte date a domande precedenti.

Per consentire un ordinamento immediato delle date ed effettuarne i controlli senza attendere input successivi, conviene in primo luogo disporre le domande nell'ordine anno,mese,giorno; quindi si debbono predisporre i controlli sull'anno, sul mese, sul giorno immessi dall'operatore per impedirgli di immettere un anno oltre il rango degli anni accettabili, o per impedirgli di immettere un mese o un giorno inesistenti.

Un programma (una subroutine) di controllo della data può realizzarsi nelle fasi seguenti:

a) richiesta di input dell'anno e accettazione all'interno di un insieme definito dal programmatore;

b) calcolo dei giorni del mese di febbraio in relazione allo anno dato in input;

c) assegnazione ad una variabile multipla G%() dei giorni di ogni mese;

d) richiesta del mese e accettazione all'interno dell'insieme 1..12

e) richiesta del giorno e accettazione all'interno degli insiemi 1..numero massimo dei giorni di ciascun mese.

Spesso si pongono dei problemi di ordinamento delle date e di calcolo dei giorni che intercorrono tra due date. Dette operazioni sono facilitate da procedure di conversione delle date in numeri ordinali che, per differenze, danno immediatamente la lunghezza del periodo intercorrente tra due date.

Si aggiungono pertanto talvolta delle routine di conversione. Per una interessante tecnica di conversione si veda oltre alle routine in BASIC la formula impiegata nei conti correnti realizzati con il MULTIPLAN.

Si utilizzi la seguente tavola di variabili:

Variabili impiegate nella subroutine di controllo data

NOME	TIPO	CONTENUTO	CARATTERI
ANNO%	INTERO	ANNO IMMESSO	2
G%()	INTERO	NUMERO DEI GIORNI DI UN MESE	2
GIORD	S.PREC	DATA CONVERTITA IN ORDINALE	4
GIORNO%	INTERO	GIORNO IMMESSO	2
MESE%	INTERO	MESE IMMESSO	2

Figura 4.1

Il programma richiede le seguenti istruzioni:

```
10100 REM SUBROUTINE DI CONTROLLO DATA
10110 NCAR%(C)=2:TIPO%(C)=1:H%(C)=H%(C)+2 LOCATE
V%(C), H%(C), 0:PRINT "ANNO":H%(C) = H%(C) +4 : GOSUB 11200:
H%(C)=H%(C)+4
```

Con queste istruzioni si visualizza la richiesta della data e si richiama la subroutine di input che inizia da riga 11200.

```
10120 IF IN$(5)="" AND GIORD>0 THEN RETURN ELSE ANNO=VAL(IN$(5))
```

Con la riga precedente, se l'operatore schiaccia il tasto di RETURN ed esiste nella variabile GIORD una data già convertita in numero ordinale, il programma accetta questa data, altrimenti prende come data l'input dell'operatore.

```
10130 IF ANNO>99 OR ANNO<50 THEN H%(C)=H%(C)-8: GOTO 10110
10140 IF ANNO MOD 4=0 THEN G%(2)=29 ELSE G%(2)=28
```

Con queste istruzioni il programma definisce il range di validità dell'anno che può dare in input l'operatore; quindi stabilisce se l'anno dato è un anno bisestile oppure no, calcolando il resto della divisione per quattro del valore di ANNO e assegna di conseguenza il valore massimo di 28 o 29 al mese di febbraio.

4.3.2 Il controllo della PARTITA IVA.

Il numero di partita IVA è costituito da un'espressione numerica di 11 cifre, di cui le prime 7 rappresentano il numero di matricola assegnato nella provincia, le successive 3 rappresentano il codice identificativo dell'ufficio IVA provinciale e l'undicesima ha funzione di flag di regolarità.

Detto flag di regolarità ha valore 0 se corrisponde al risultato della seguente verifica:

"Si moltiplicano per due le cifre di posizione pari (2', 4', 6', 8', 10') a partire da sinistra del campo IVA.

Tutte le singole cifre così ottenute vanno poi sommate tra loro e con le cifre di ordine dispari (1', 3', 5', 7', 9') del numero di partita.

Nel caso che l'ultima cifra del risultato ottenuto sia zero, il codice di controllo del numero di partita IVA deve essere zero, altrimenti deve essere uguale al complemento a 10 della suddetta ultima cifra.

Va ulteriormente impostato ad 1 se i primi sette caratteri della partita IVA sono 0 e se dal quart'ultimo al penultimo carattere non è indicato un numero compreso tra 001 e 095". Gazzetta Ufficiale n. 37 8 Febbraio 1982.

Il listato della subroutine per il controllo della IVA è presentato a pag. 55.

4.3.3 Il controllo del codice fiscale.

Il codice fiscale delle persone fisiche è costituito da un'espressione alfanumerica di cui i primi 15 caratteri costituiscono la codifica dei dati anagrafici della persona ed il sedicesimo ha funzione di controllo.

Il codice fiscale degli altri soggetti coincide con il numero di partita IVA.

Il codice fiscale è imposto dal decreto ministeriale del 23/12/1976 che stabilisce quanto segue:

LEGGI E DECRETI

DECRETO MINISTERIALE 23 dicembre 1976.

Sistemi di codificazione dei soggetti da iscrivere all'anagrafe tributaria.

IL MINISTRO PER LE FINANZE

Visto il decreto del Presidente della Repubblica 29 settembre 1973, n. 605, concernente disposizioni relative all'anagrafe tributaria ed al codice fiscale dei contribuenti;

Visto l'art. 2, comma primo, del decreto del Presidente della Repubblica 2 novembre 1976, n. 784, recante modificazioni ed integrazioni al citato decreto del Presidente della Repubblica 29 settembre 1973, n. 605;

Considerato che si rende necessario stabilire i sistemi di codificazione da adottare per la iscrizione all'anagrafe tributaria delle persone fisiche e dei soggetti diversi dalle persone fisiche;

Decreta:

Art. 1.

Sistemi di codificazione

Le persone fisiche, le persone giuridiche e le società, associazioni ed altre organizzazioni di persone o di beni prive di personalità giuridica sono iscritte all'anagrafe tributaria secondo appositi sistemi di codificazione.

Art. 2.

Numero di codice fiscale delle persone fisiche

Il numero di codice fiscale delle persone fisiche è costituito da una espressione alfanumerica di sedici caratteri.

I primi quindici caratteri sono indicativi dei dati anagrafici di ciascun soggetto secondo l'ordine seguente:

tre caratteri alfabetici per il cognome;

tre caratteri alfabetici per il nome;

due caratteri numerici per l'anno di nascita;

un carattere alfabetico per il mese di nascita;

due caratteri numerici per il giorno di nascita ed il sesso;

quattro caratteri (uno alfabetico e tre numerici) per il comune italiano o per lo Stato estero di nascita.

Il sedicesimo carattere, alfabetico, ha funzione di controllo.

Art. 3.

Caratteri indicativi del cognome

I cognomi che risultano composti da più parti o comunque separati od interrotti, vengono considerati come se fossero scritti secondo un'unica ed ininterrotta successione di caratteri.

Per i soggetti di sesso femminile coniugati si prende in considerazione soltanto il cognome da nubile.

Se il cognome contiene tre o più consonanti, i tre caratteri da rilevare sono, nell'ordine, la prima, la seconda e la terza consonante.

Se il cognome contiene due consonanti, i tre caratteri da rilevare sono, nell'ordine, la prima e la seconda consonante e la prima vocale.

Se il cognome contiene una consonante e due vocali, si rilevano, nell'ordine, quella consonante e quindi la prima e la seconda vocale.

Se il cognome contiene una consonante e una vocale, si rilevano la consonante e la vocale, nell'ordine, e si assume come terzo carattere la lettera x (ics).

Se il cognome è costituito da due sole vocali, esse si rilevano, nell'ordine, e si assume come terzo carattere la lettera x (ics).

Art. 4.

Caratteri indicativi del nome

I nomi doppi, multipli o comunque composti, vengono considerati come scritti per esteso in ogni loro parte e secondo un'unica ed ininterrotta successione di caratteri.

Se il nome contiene quattro o più consonanti i tre caratteri da rilevare sono, nell'ordine, la prima, la terza e la quarta consonante.

Se il nome contiene tre consonanti, i tre caratteri da rilevare sono, nell'ordine, la prima, la seconda e la terza consonante.

Se il nome contiene due consonanti, i tre caratteri da rilevare sono, nell'ordine, la prima e la seconda consonante e la prima vocale.

Se il nome contiene una consonante e due vocali, i tre caratteri da rilevare sono, nell'ordine, quella consonante e quindi la prima e la seconda vocale.

Se il nome contiene una consonante e una vocale, si rilevano la consonante e la vocale, nell'ordine, e si assume come terzo carattere la lettera x (ics).

Se il nome è costituito da due sole vocali, esse si rilevano nell'ordine, e si assume come terzo carattere la lettera x (ics).

Art. 5.

Data, sesso e luogo di nascita

I due caratteri numerici indicativi dell'anno di nascita sono, nell'ordine, la cifra delle decine e la cifra delle unità dell'anno stesso.

Il carattere alfabetico corrispondente al mese di nascita è quello stabilito per ciascun mese nella seguente tabella:

Gennaio	= A	Maggio	= E	Settembre	= P
Febbraio	= B	Giugno	= H	Ottobre	= R
Marzo	= C	Luglio	= L	Novembre	= S
Aprile	= D	Agosto	= M	Dicembre	= T

I due caratteri numerici indicativi del giorno di nascita e del sesso vengono determinati nel modo seguente:

per i soggetti maschili il giorno di nascita figura invariato, con i numeri da uno a trentuno, facendo precedere dalla cifra zero i giorni del mese dall'uno al nove. Per i soggetti femminili il giorno di nascita viene aumentato di quaranta unità, per cui esso figura con i numeri da quarantuno a settantuno.

I quattro caratteri alfanumerici indicativi del comune italiano o dello Stato estero di nascita, costituiti da un carattere alfabetico seguito da tre caratteri numerici, si rilevano rispettivamente dal volume « Codice dei comuni d'Italia » o dal volume « Codice degli Stati esteri », redatti a cura della Direzione generale del catasto e dei servizi tecnici erariali.

All'aggiornamento dei volumi di cui al precedente comma provvede la Direzione generale del catasto e dei servizi tecnici erariali.

Art. 6.

Persone fisiche con identica espressione alfanumerica

Quando l'espressione alfanumerica relativa ai primi quindici caratteri del codice risulta comune a due o più soggetti, si provvede a differenziarla per ciascuno dei soggetti successivi al primo soggetto codificato. A tal fine, si effettuano, nell'ambito dei sette caratteri numerici contenuti nel codice, sistematiche sostituzioni di una o più cifre a partire da quella di destra, con corrispondenti caratteri alfabetici secondo la seguente tabella:

0 = L	5 = R
1 = M	6 = S
2 = N	7 = T
3 = P	8 = U
4 = Q	9 = V

Art. 7.

Carattere alfabetico di controllo

Il sedicesimo carattere ha funzione di controllo della esatta trascrizione dei primi quindici caratteri. Esso viene determinato nel modo seguente: ciascuno degli anzidetti quindici caratteri, a seconda che occupi posizione di ordine pari o posizione di ordine dispari, viene convertito in un valore numerico in base alle corrispondenze indicate rispettivamente ai successivi punti 1) e 2).

1) Per la conversione dei sette caratteri con posizione di ordine pari:

A o zero	= zero	O	= 14
B o 1	= 1	P	= 15
C o 2	= 2	Q	= 16
D o 3	= 3	R	= 17
E o 4	= 4	S	= 18
F o 5	= 5	T	= 19
G o 6	= 6	U	= 20
H o 7	= 7	V	= 21
I o 8	= 8	W	= 22
J o 9	= 9	X	= 23
K	= 10	Y	= 24
L	= 11	Z	= 25
M	= 12	—	—
N	= 13	—	—

2) Per la conversione degli otto caratteri con posizione di ordine dispari:

A o zero	= 1	O	= 11
B o 1	= 0	P	= 3
C o 2	= 5	Q	= 6
D o 3	= 7	R	= 8
E o 4	= 9	S	= 12
F o 5	= 13	T	= 14
G o 6	= 15	U	= 16
H o 7	= 17	V	= 10
I o 8	= 19	W	= 22
J o 9	= 21	X	= 25
K	= 2	Y	= 24
L	= 4	Z	= 23
M	= 18	—	—
N	= 20	—	—

I valori numerici così determinati vengono addizionati e la somma si divide per il numero 26.

Il carattere di controllo si ottiene convertendo il resto di tale divisione nel carattere alfabetico ad esso corrispondente nella sottoindicata tabella:

zero	= A	14	= O
1	= B	15	= P
2	= C	16	= Q
3	= D	17	= R
4	= E	18	= S
5	= F	19	= T
6	= G	20	= U
7	= H	21	= V
8	= I	22	= W
9	= J	23	= X
10	= K	24	= Y
11	= L	25	= Z
12	= M	—	—
13	= N	—	—

Art. 8.

Numero di codice fiscale dei soggetti diversi dalle persone fisiche

Il numero di codice fiscale dei soggetti diversi dalle persone fisiche è costituito da una espressione numerica di undici cifre. Le prime sette cifre rappresentano il numero di matricola del soggetto nell'ambito della provincia in cui ha sede l'ufficio che attribuisce il numero di codice fiscale; esso si ottiene, per ciascun soggetto, incrementando di una unità il numero di matricola stabilito per il soggetto che immediatamente lo precede.

Le tre cifre dall'ottava alla decima rappresentano il codice identificativo della provincia in cui ha sede l'ufficio che attribuisce il numero di codice fiscale.

L'undicesimo carattere ha funzione di controllo dell'esatta trascrizione delle prime dieci cifre.

Art. 9.

Carattere numerico di controllo

Il carattere di controllo viene determinato nel modo seguente:

si sommano i valori di ciascuna delle cinque cifre di ordine dispari, partendo da sinistra;

si raddoppia ogni cifra di ordine pari e, se il risultato è un numero di due cifre, esso si riduce ad una sola sommando la cifra relativa alle decine e quella relativa alle unità; si sommano quindi tutti i precedenti risultati;

si determina il totale delle due somme di cui sopra;

si sottrae da dieci la cifra relativa alle unità del precedente totale. Il carattere di controllo è la cifra relativa alle unità del risultato.

Art. 10.

Numero di codice fiscale provvisorio

L'Amministrazione finanziaria può attribuire un numero di codice fiscale provvisorio.

Il numero di codice fiscale provvisorio delle persone fisiche ha struttura e composizione uguali a quelle di cui al precedente art. 8. Le prime sette cifre rappresentano il numero di matricola del soggetto; le cifre dall'ottava alla decima identificano l'ufficio che attribuisce il numero di codice fiscale provvisorio e l'undicesima è il carattere di controllo, che viene determinato con le modalità di cui all'art. 9.

Ha inoltre validità di numero di codice fiscale provvisorio il numero di codice fiscale attribuito a soggetti persone fisiche, avente struttura e composizione conformi agli articoli da 2 a 7 del presente decreto, in corrispondenza del quale siano errati uno o più dati anagrafici che concorrono alla formazione del numero di codice fiscale stesso. Si applicano in tal caso le disposizioni di cui agli articoli 5, secondo comma, e 19 del decreto del Presidente della Repubblica 2 novembre 1976, n. 784.

Il numero di codice fiscale provvisorio dei soggetti diversi dalle persone fisiche ha struttura uguale a quella del numero di codice fiscale definitivo.

Il presente decreto sarà pubblicato nella *Gazzetta Ufficiale* della Repubblica italiana.

Roma, addì 23 dicembre 1976

Il Ministro: PANDOLFI

(13813)

Il listato della subroutine per il controllo del codice fiscale è presentato a pag. 54.

CAPITOLO 5

LA ESECUZIONE DI MODULI DI PROGRAMMA GESTITA DA MENU'

5.1. La esecuzione di moduli di programma gestita da menù:
5.1.1 la costruzione di menù generali;
5.1.2 la costruzione di menù specifici;
5.1.3 i menù di un programma di contabilità generale.

5.1 La esecuzione di moduli di programma gestita da menù

Un programma facile da usare deve avere caratteristiche di elasticità di funzionamento tali da consentirgli di adattarsi il più possibile alle esigenze dell'operatore.

Il sistema informativo generale è composto di sottosistemi che corrispondono alle funzioni aziendali fondamentali, ad esempio le seguenti:

1) Gestione vendite
2) Clienti
3) Gestione magazzino
4) Gestione acquisti
5) Fornitori
6) Contabilità generale
7) Paghe
8) Gestione della produzione
9) Statistiche.

Dette funzioni siano realizzate mediante programmi scomposti in moduli a esecuzione in successione non vincolata.

Disponga l'azienda di una rete di calcolatori con funzioni definite; ad esempio ad un terminale siano assegnate le registrazioni degli ordini, la fatturazione e la contabilità clienti.

Si sostiene che un programma è di facile e comodo impiego se l'operatore può chiedere la esecuzione di una delle tre procedure e, all'interno di queste, di un modulo particolare piuttosto che di un altro.

Per detto scopo deve essere costruito un sistema di menù che faccia conoscere all'operatore su quali opzioni può contare e che gli consenta di metterle facilmente in esecuzione con le limitazioni connesse alle funzioni specifiche assegnate ad un singolo terminale.

5.1.1 La costruzione di menù generali.

Il menù generale deve offrire all'operatore sia la possibilità di mettere in esecuzione tutti i moduli di sistema, che quella di mettere in esecuzione delle utility generali quali la copia dei dischi o dei dischetti, la copia di file, il controllo dei dati e le opportune correzioni. Detto menù offre in sostanza una serie di opzioni numerate che è possibile mettere in esecuzione dandone da tastiera il numero corrispondente.

Costruito un video che visualizzi il menù, secondo le già viste istruzioni di pag. 20 e richiesta la risposta dell'operatore, la esecuzione di un dato modulo viene eseguita mediante istruzioni di rinvio a righe di richiamo dei moduli di programmi da mettere in esecuzione.

La istruzione di rinvio alle righe di richiamo dei moduli di possibile esecuzione sulla base della risposta dell'operatore.

```
ON...GOTO.....
```

La istruzione di richiamo ed esecuzione di un modulo di programma esterno a quello nel momento in esecuzione è invece:

```
CHAIN "NOME PROGRAMMA"
```

Le istruzioni in concreto potrebbero essere le seguenti:

```
100 REM MAIN PROGRAM
110 GOSUB 18000' PER IL MENU' PRINCIPALE
120 C=11:GOSUB 11200'PER L'INPUT DI RISPOSTA
130 AN=VAL(IN$(5))
140 ON AN GOTO 150,160,170,180,190,200,210,220,230,240,250
150 CHAIN"UTILITY"
160 CHAIN"GESVEN"
170 CHAIN"MAG"
180 CHAIN"CLIENTI"
190 CHAIN"GESAC"
200 CHAIN"FORN"
210 CHAIN"GEN"
220 CHAIN"PAGHE"
230 CHAIN"PROD"
240 CHAIN"STAT"
250 STOP
```

5.1.2 La costruzione di menù specifici.

Come è lasciata all'operatore la scelta tra diverse applicazioni, così per ogni applicazione gli può essere lasciata la scelta di esecuzione di diversi moduli di programma. Dette opzioni vengono presentate mediante appositi menù. Altri menù specifici si

possono incontrare durante lo sviluppo della applicazione, eventualmente alternati con normali videate e stampe.

La tecnica di costruzione non differisce sensibilmente dalla tecnica di costruzione di un menù generale. In genere si può affermare che per i menù specifici la successione delle istruzioni ON... GOTO e CHAIN con eventuale richiesta di altri dischetti, può essere sostituita dalla istruzione diretta ON... GOSUB

5.1.3 I menù di un programma di contabilità generale.

Un programma di contabilità generale, per la parte inerente la apertura dei conti potrebbe avere menù e video organizzati nella struttura seguente (vedi figure da 5.1 a 5.6):

```
-----------------------------------------------------
|                MENU' GENERALE                      |
|                                                    |  In caso di
|     1 - Utility e archivi di base                  |  1  CHAIN "UTILITY"
|     2 - Gestione vendite                           |  2  CHAIN "GESVEN"
|     3 - Magazzino                                  |  3  CHAIN "MAG"
|     4 - Clienti                                    |  4  CHAIN "CLIENTI"
|     5 - Gestione acquisti                          |  5  CHAIN "GESAC"
|     6 - Fornitori                                  |  6  CHAIN "FORN"
|     7 - Contabilita' generale                      |  7  CHAIN "GEN"
|     8 - Paghe                                      |  8  CHAIN "PAGHE"
|     9 - Produzione                                 |  9  CHAIN "PROD"
|    10 - Statistiche                                | 10  CHAIN "STAT"
|    11 - FINE LAVORO                                | 11  STOP
|                     Dammi la tua opzione           |
|____________________________________________________|
```

Figura 5.1

```
-----------------------------------------------------
|        MENU' DI CONTABILITA' GENERALE              |  In caso di
|                                                    |  va al video
|     1 - Creazione apertura conti                   |              7.1
|     2 - Registrazione movimenti contabili          |              7.2
|     3 - Estratti conto                             |              7.3
|     4 - Giornale                                   |              7.4
|     5 - Bilanci di verifica                        |              7.5
|     6 - Chiusura                                   |              7.6
|     7 - Stato Patrimoniale                         |              7.7
|     8 - Profitti e Perdite                         |              7.8
|     9 - FINE LAVORO                                |  Torna al master menu'
|                                                    |
|                                                    |
|                     Dammi la tua opzione           |
|____________________________________________________|
   Video 7.1
```

Figura 5.2

```
-----------------------------------------------------
|                 CONTABILITA' GENERALE              |
|                                                    |
|     1 - Creazione apertura conti                   |
|                                                    |
|   DAMMI LA DATA __/__/__/                          | A
|                                                    |
|                                                    |
|                                                    |
|   DAMMI IL NOME DEL FILE  _________________        | B
|                                                    |
|                                                    |
|                                                    |
|                     Confermi  (Y/N)                | n torna ad A
|                                                    | Y continua
|____________________________________________________|
    Video 7.1.1
```

Figura 5.3

```
-----------------------------------------------------
|                 CONTABILITA' GENERALE              |
|                                                    |
|     1 - Creazione apertura conti                   |
|                                                    |
|   CONTO DI GRUPPO N.                               | A
|                                                    |
|                                                    |
|                                                    |
|   DAMMI IL NOME DEL CONTO ______________________   | B
|   ______________________________________________   |
|                                                    |
|                         Confermi (Y/N)             | N torna a B
|                                                    | Y continua
|                              Un altro (Y/N)        | Y torna ad A
|____________________________________________________| N continua
    Video 7.1.2
```

Figura 5.4

```
-----------------------------------------------------
|                 CONTABILITA' GENERALE              |
|                                                    |
|     1 - Creazione apertura conti                   |
|                                                    |
|   CONTO DI GRUPPO N.XY                             | A
|   DENOMINAZIONE XXXXXXXXXXXXXXXXXXXXXXXXXXXX       |
|                                                    |
|                                                    |
|   DAMMI IL NOME DEL CONTO DI MASTRO ___________    | B
|   ______________________________________________   |
|                                                    |
|                         Confermi (Y/N)             | N torna a B
|                                                    | Y continua
|                              Un altro (Y/N)        | Y torna ad A
|____________________________________________________| N continua
    Video 7.1.3
```

Figura 5.5

```
-----------------------------------------------------
|                CONTABILITA' GENERALE                |
|                                                     |
|        Creazione apertura conti                     |
|                                                     | In caso di
|   1 - Visualizzazione Indice                        | 1 va al video 7.1.5
|                                                     |
|   2 - Stampa Indice                                 | 2 va al video 7.1.6
|                                                     |
|   3 - Stampa lista piano dei conti                  | 3 va al video 7.1.7
|                                                     |
|   4 - FINE LAVORO                                   | 4 torna al video 7.1
|                                                     |
|                    Dammi la tua opzione             |
|                                                     |
|-----------------------------------------------------|
   Video 7.1.4
```

Figura 5.6

CAPITOLO 6

CARATTERI DEGLI ARCHIVI GENERALI NEL SISTEMA INFORMATIVO AZIENDALE

6.1 Gli archivi generali e i caratteri di stabilità e accessibilità dei dati in essi archiviati.
6.2 Il Profile: la sua funzione, la sua struttura.
6.3 I programmi di gestione di un profile e degli altri archivi.

6.1 Gli archivi generali e i caratteri di stabilità e accessibilità dei dati in essi archiviati.

L'utilizzo dei dati archiviati consente di distinguere tra due tipi di archivio: **archivi permanenti** e **archivi temporanei**. I primi sono destinati a contenere dati di base essenziali quali quelli relativi alle anagrafiche dei clienti, dei fornitori e alle informazioni generali sulla azienda che permangono con poche variazioni nel tempo; i secondi sono destinati ai dati che servono per gli aggiornamenti periodici degli archivi permanenti o di speciali archivi riassuntivi; si tratta ad esempio dei dati dei movimenti dei conti correnti e di contabilità che, una volta impiegati per i relativi aggiornamenti, possono essere eliminati.

Notiamo che per questi archivi temporanei la successione temporale di registrazione coincide con quella logica per cui si adotta di solito una organizzazione sequenziale. Per gli archivi permanenti, invece, deve essere consentito un accesso diretto senza una logica prestabilita per cui si adottano organizzazioni di dati dirette o per chiave o con indici. Si ricorda che in una organizzazione sequenziale è possibile reperire i dati attraverso una ricerca sequenziale, esaminando i record dal primo all'ultimo, che in una organizzazione diretta i dati possono essere reperiti mediante indicazione diretta dell'indirizzo del record in cui sono contenuti, che in una organizzazione ad indice i dati possono essere ritrovati ricorrendo ad una tabella in cui vi sia corrispondenza tra una chiave ed il record da ricercare.

La **creazione** richiede una attenta definizione della lunghezza dei record e della struttura dei campi che tenga conto delle caratteristiche dei dati da archiviare e delle esigenze di utilizzazione. Pertanto devono essere definiti i collegamenti tra i diversi archivi della azienda, ma anche devono essere facilitati gli accessi ai file con record a funzioni specifiche diversificate, eventualmente con la predisposizione di indici composti di descrizioni di record e di puntatori.

Per ogni tipo di archivio deve essere assicurata possibilità di **modifica** oltre che di creazione.

I dati da archiviare in creazione o modifica debbono essere controllati all'atto della immissione nell'archivio stesso. Il controllo deve riguardare il tipo dei dati immessi rispetto a quello richiesto, il numero dei caratteri dei dati immessi e la loro coerenza.

Se un archivio oltre che permanente è anche generale, cioè serve a più applicazioni, il suo accesso deve essere necessariamente consentito, oltre che al programma di creazione stesso anche, almeno in lettura, ai programmi che ne abbisognano.

L'accesso ai file può essere consentito per raccogliere grandi insiemi di informazioni o per leggere e eventualemte modificare i dati di uno o più record.
Per soddisfare la prima esigenza è necessario approntare programmi di ordinamento, lettura e visualizzazione o stampa delle informazioni; per la seconda è necessario approntare programmi di ricerca, lettura, aggiornamento di record.

6.2.Il Profile: la sua funzione, la sua struttura.

Notiamo che alcune informazioni generali sulla azienda (ragione sociale, ditta, codice fiscale, partita I.V.A., capitale sottoscritto e versato etc.) potrebbero essere assegnate a costanti direttamente nei programmi che le utilizzano. Nello stesso modo potrebbero essere assegnati i dati da utilizzare in molte applicazioni specifiche per la definizione in chiaro di codici e causali, per la elencazione delle diverse aliquote I.V.A., per la definizione degli scaglioni I.R.P.E.F. etc..

Un simile modo di procedere presenta degli inconvenienti soprattutto nel momento di una eventuale rettifica, da farsi necessariamente per tutti i programmi che utilizzano l'informazione modificata.

Per evitare simili inconvenienti, riducendo inoltre la quantità di memoria centrale occupata, si preferisce registrare tali informazioni in un apposito archivio generale che, siccome serve alla gestione del sistema nel suo insieme e agli altri file, si chiama "profile".

Il profile accoglie dati costanti o relativamente tali che servono a più di una applicazione. Talvolta accoglie anche i dati inerenti le caratteristiche dei file, le tabelle dei codici e delle causali delle applicazioni specifiche. Mediante il profile si dà in sintesi una prima descrizione della azienda e del suo sistema informativo, riunendo in esso i dati delle sue caratteristiche generali.

Il profile è di solito costituito dei seguenti elementi:

1) un indice per la gestione del profile stesso
2) i dati generali della azienda
3) i dati dei file delle singole procedure e delle loro tabelle codici-casuali.

Si ritiene che l'indice e le informazioni di profile possano essere accolte in due distinti file di record . Si scelga per la struttura record dell'indice un record costituito di un campo numerico di 10 caratteri e di tre campi alfabetici di 2 caratteri ciascuno per accogliere le chiavi di ciascun record e dei valori numerici interi. Questi campi numerici nel primo record sono destinati alla gestione dell'indice stesso. Accoglieranno infatti il numero dei record di indice registrati, il numero dei record cancellati e in attesa di riutilizzo, il numero dei record di profile utilizzati.

I successivi record di indice accoglieranno invece la chiave di ciascun record di profile, l'indirizzo del record di indice, un campo vuoto e un puntatore al record di profile che corrisponde alla chiave in indice.

L'insieme dei record di indice costituisce appunto un indice che per pagine di stampa di videate può essere presentato nel modo seguente (fig 6.1) :

```
Indice

    |--------------|---------------|---------------|---------------|
    | Titolo       |Record pieni I.|Record vuoti I.|Record pieni P.|
    |--------------|---------------|---------------|---------------|
  1 |              |               |               |               |
    |--------------|---------------|---------------|---------------|

    | Chiave       | (indirizzo)   |               | Punt. Profile |
    |--------------|---------------|---------------|---------------|
  2 |              |               |               |               |
    |--------------|---------------|---------------|---------------|
  3 |              |               |               |               |

  n |              |               |               |               |
    |--------------|---------------|---------------|---------------|
```

Figura 6.1

Ciascun record di profile nell'esempio è costituito di 256 caratteri divisi in tre campi numerici per valori interi da 2 caratteri ciascuno, e 10 campi numerici da 20 caratteri ciascuno più 10 campi numerici da 4 caratteri per accogliere causali e descrizioni e relativi codici delle tabelle da utilizzare nelle specifiche applicazioni.

Un record di profile destinato ad accogliere la tabella delle causali e dei codici delle movimentazioni di contabilità generali avrà pertanto il seguente contenuto.

```
|------------------|-------------------|-------------------|
| Puntatore indice | Puntatore seguito | numero campi pieni|
|------------------|-------------------|-------------------|
|__________________|___________________|___________________|

          ___________________________________
          | Codici      | Causali            |
          |-------------|--------------------|
         1|_____________|____________________|
         2|_____________|____________________|
         3|_____________|____________________|
         4|_____________|____________________|
         5|_____________|____________________|
         6|_____________|____________________|
         7|_____________|____________________|
         8|_____________|____________________|
         9|_____________|____________________|
        10|_____________|____________________|
```

Figura 6.2

6.3. I programmi di gestione di un profile e degli altri archivi

L' uso del profile, come l'uso di qualsiasi tipo di file, richiede un programma di creazione e di modifica della sua struttura, di accesso per lettura registrazione e modifica di dati particolari.

Possiamo pertanto ritenere che un programma di gestione file sia composto dei seguenti moduli:

Crea
Aggiungi
Cerca
Modifica
Cancella
Copia

Disponendo della vista struttura dati il programma di creazione può essere realizzato seguendo la seguente pseudocodifica:

6.3.1 Sottoprogramma CREA.

Metti a 1 il numero dei record pieni
metti a 0 il numero dei record vuoti
il numero dei record di profile

RIPETI
INCREMENTA il numero dei record indice pieni
INCREMENTA il numero dei record di profile
INPUT dati del record di indice al numero di record indice REGISTRA
INPUT dati del record di profile al numero di record di profile REGISTRA

FINCHE' non hai finito.

Mettiti sul record numero 1 e REGISTRA il numero dei record pieni, il numero di quelli vuoti e il numero dei record di profile.

Quindi TORNA al programma principale.

La aggiunta di un nuovo record di profile può essere realizzata seguendo la seguente pseudocodifica:

6.3.2 Sottoprogramma AGGIUNGI.

RIPETI

LEGGI il primo record di indice

SE il numero dei record cancellati è maggiore di 1 TOGLI 1 dai record cancellati

ALTRIMENTI AGGIUNGI uno al puntatore dei record di profile;
REGISTRA il puntatore al profile all'indirizzo corrispondente al numero dei record pieni

Mettiti sul primo record e REGISTRA i cambiamenti

Mettiti sull'ultimo dei record di indice
IMMETTI e
REGISTRA i dati di indice

Mettiti sul record di profile puntato
IMMETTI e
REGISTRA i dati relativi

FINCHE' non sia finito

Quindi RIORDINA
TORNA al programma principale.

La cancellazione di un record di profile viene fatta togliendolo dall'indice pur tenendolo a disposizione nel profile per una nuova registrazione.

6.3.3 Sottoprogramma CANCELLA.

Sottoprogramma CANCELLA

SOSTITUISCI la chiave con "zzz" al record di indice da togliere e REGISTRA

al record 1 AGGIUNGI 1 al numero dei record vuoti.
TOGLI 1 al numero dei record pieni
e REGISTRA.

RITORNA al programma principale che deve riordinare.

Si danno qui di seguito i menù e i video-formattati per la gestione del profile. Gli stessi menù e le stesse videate potrebbero essere impiegate per la gestione di tutti gli archivi. (Vedi figure da 6.3 a 6.8)

```
-----------------------------------------------------
|                                                   |
|                        PROFILE                    |
|                                                   |
|    1 - CREAZIONE                                  |
|                                                   |
|                                                   |
|                                                   |
|    2 - UTILIZZO                                   |
|                                                   |
|                                                   | In caso di
|                                                   |
|                                                   |
|              Dammi la tua opzione (1) o (2)       | 1 va al video 1.1
|                                                   | 2 va al video 1.2
|-----------------------------------------------------|
                    MENU' 1
```

Figura 6.3

```
-----------------------------------------------------
|                                                   |
|                        PROFILE                    |
|                                  - CREAZIONE -    |
|                                                   |
|    Dammi il nome del file                         | A
|                                                   |
|                                                   |
|                                                   |
|    Dammi il titolo dell'indice                    | B
|                                                   |
|                              O K ? (Y) O (N)      | N torna ad A
|                                                   | Y continua con
|                                                   | il video 1.1.2
|                                                   |
|-----------------------------------------------------|
            Video 1.1.1
```

Figura 6.4

```
-----------------------------------------------------
|                                                   |
|                        PROFILE                    |
|                                  - CREAZIONE -    |
|                                                   |
|    Dammi il Titolo della TABELLA                  | A
|                                                   |
|                                                   |
|                                                   |
|                                                   |
|                                                   |
|                              O K ? (Y) O (N)      | N torna ad A
|                                                   | Y continua con
|                                                   | il video 1.1.3
|                                                   |
|-----------------------------------------------------|
            Video 1.1.2
```

Figura 6.5

```
-----------------------------------------------------
|                                                   |
|                        PROFILE                    |
|     CODICI IVA                                    |
|    CAUSALI                          CODICI        |
|          8%                              1        |
|         10%                              2        |
|         18%                              3        |
|         38%                              4        |
|  Non imponibile a.8                      5        |
|  Non imponibile a.8.b                    6        |
|  Non imponibile a.9                      7        |
|  Esente a. 10                            8        |
|  Escluso a.15                            9        |
|                 ALTRA TABELLA ? (Y) o (N)         | Y torna al video 1.1.2
|___________________________________________________| N Fine lavoro
           Video 1.1.3
```

Figura 6.6

```
-----------------------------------------------------
|                                                   |
|                        PROFILE                    |
|                                UTILIZZO           |  In caso di
| 1 - Visualizza indice                             |  1 va al video 1.2.2
|                                                   |
| 2 - Ricerca record (modifica, cancella, accetta)  |  2 va al video 1.2.3
|                                                   |
| 3 - Aggiungi e riordina                           |  3 va al video 1.2.4
|                                                   |
| 4 - Fine lavoro                                   |  4 torna al menu' generale
|                                                   |
|                                                   |
|                Dammi la tua opzione               |
|                                                   |
|___________________________________________________|
           Video 1.2.1
```

Figura 6.7

```
-----------------------------------------------------
|              Indice di PROFILE                    |
|                                                   |
|              rec. pieni-cancellati-pointer prof.  |
|  Profindex             6          3           8   |
|  CHIAVE           INDIRIZZO           PUNTATORE   |
|  Codici IVA            2                       1  |
|  Movimenti contabilita'3                       2  |
|  Movimenti magazzino   4                       5  |
|  Scadenze  pagamenti   5                       4  |
|  Tipo IVA              6                       8  |
|  ZZZ                   7                       3  | In caso di
|  ZZZ                   8                       7  | + scorre in avanti
|  ZZZ                   9                       6  | - scorre indietro
|                        Avanti (+) Indietro (-)    | Se e' finito torna al
|___________________________________________________| menu'
          Video 1.2.2
```

Figura 6.8

CAPITOLO 7

GLI ARCHIVI SPECIFICI

7.1. I master file.
7.2. I file transazionali.
7.3. I file riassuntivi.
7.4. I collegamenti tra i diversi file.
7.5 I modi di gestione dei file.

7.1 I master file

Si possono dire archivi specifici quelli creati da una specifica applicazione per le sue dirette necessità. Si pensi ai file di anagrafica clienti e fornitori creati rispettivamente dalle procedure clienti e fornitori, o alla anagrafica dipendenti etc.

Si fa notare che non è detto che questi archivi debbano e possano essere utilizzati solo da quelle applicazioni che li hanno creati, infatti, se spesso le possibilità di lettura e registrazione sono esclusive, possibilità di lettura vengono lasciate anche ad altre applicazioni.

Come per gli archivi generali le informazioni raccolte negli archivi specifici possono avere carattere **permanente**. Si tratta in genere di informazioni anagrafiche sui clienti, sui fornitori, sui dipendenti, di dati del piano dei conti utilizzato in contabilità generale, di dati di anagrafica degli agenti e dei rappresentanti etc. I file che raccolgono dette informazioni sono anche detti con termine inglese master-file.

7.2 I file transazionali

Alcuni archivi sono destinati ad accogliere informazioni che riguardano le singole operazioni aziendali e le rappresentano così come si succedono nel tempo. Rispetto alle informazioni di master file queste sono volte a rappresentare la dinamica dello svolgimento della attività aziendale attraverso gli aspetti elementari delle operazioni compiute. Gli archivi che accolgono simili informazioni si dicono transazionali. Appartengono ai file transazionali i file dei movimenti di magazzino, degli ordini a fornitori, degli ordini da clienti, dei movimenti di giornale etc.

7.3 I file riassuntivi

Dai file transazionali si traggono spesso file da utilizzare in procedure diverse da quella di creazione del file di transazione stesso. Ad esempio dal file fatture della procedura fatturazione si potranno ottenere informazioni in lettura per le procedure di magazzino e fornitori per il controllo delle differenze tra ordinato, fatturato e consegnato. Dette informazioni talvolta per sovrabbondanza possono essere registrate in appositi file di comunicazione.

Dai file transazionali si possono trarre file di dati da stampare a prova delle operazioni elementari compiute. Si ricordi che la lentezza delle stampanti rispetto alla velocità delle altre periferiche rende opportuna la predisposizione di appositi file di stampa da eseguirsi in tempi successivi a quelli in cui le informazioni da stampare sono rese disponibili dalle unità di lettura ed elaborazione.

Inoltre si nota che disposizioni legislative impongono la tenuta di giornali (vedansi pagg. 59 e seguenti) in cui sia data prova della successione cronologica delle operazioni compiute.

Dai file transazionali temporanei si deducono anche dati di sintesi da registrare in appositi file riassuntivi che rilevano aspetti complessi della realtà aziendale. Si tratta, ad esempio, delle informazioni inerenti gli effetti dei singoli movimenti di credito e di debito di ciascun cliente o fornitore sui loro saldi a credito o a debito; si tratta, ad esempio, dei raggruppamenti dei valori dei movimenti del medesimo tipo quali le riclassificazioni dei movimenti dei sottoconti nei conti di mastro e dei movimenti di questi ultimi nei conti di gruppo.

Si nota che un estratto conto è sempre composto di una intestazione, con i dati anagrafici del cliente o del fornitore, di un corpo con i dati delle operazioni in corso di esecuzione, di una parte finale riassuntiva con il saldo finale ed eventualmente i saldi delle partite aperte. Si nota ancora che i dati delle informazioni sono dati stabili, mentre variabili sono i dati riassuntivi finali delle operazioni compiute. Quindi i file su cui si fonda la formazione degli estratti conto saranno i seguenti:

1) Il file di anagrafica clienti o fornitori per le intestazioni degli estratti conto;
2) Il file dei movimenti di conto per il corpo;
3) Un file riassuntivo con il saldo finale e le eventuali partite aperte.

Talvolta si ritiene opportuno rilevare i dati riassuntivi direttamente nel file di dati permanenti, ed allora mancano i file riassuntivi.

In genere questo sistema di file costituisce la struttura fondamentale di registrazione dei fatti amministrativi, rilevati nei loro aspetti elementari originari in archivi temporanei transazionali, rilevati negli aspetti più complessi in archivi riassuntivi.

7.4. I collegamenti tra i file

La presenza di più file da trattare per l'adempimento di determinate funzioni rende evidenti le esigenze dei collegamenti tra i dati dei diversi file. Questi collegamenti possono essere realizzati mediante puntatori. Le connessioni tra file di anagrafica, file riassuntivi e file di movimenti può essere realizzata nel modo seguente:

1) ciascun record di anagrafica è collegato al corrispondente record riassuntivo da un puntatore;
2) ciascun record riassuntivo è collegato al file di movimenti da un puntatore al primo record che accoglie un movimento che interessa il record riassuntivo; eventualmente un altro puntatore può collegare il record riassuntivo con l'ultimo movimento che lo riguarda;

3) nel file dei movimenti i movimenti interessanti uno stesso record riassuntivo sono collegati da due puntatori uno al record che accoglie il precedente movimento della stessa specie, un altro al successivo.

Vedasi fig.7.1 per una rappresentazione dei vincoli tra i suddetti diversi tipi di archivio.

```
-----------------------------------
!A N A G R A F I C A   C L I E N T I!
!-----------------------------------!
! Cognome Indirizzo    etc   pointer!
!-----------------------------------!
!1 Carli  via Roma  5         1     !
!2 Dani   Via Sani  8         2     !
!3 Eli    Via Zeni  7         3     !
!4 Fani   Via RENI            4     !
!5 Giorgi V_________________________________
!6 Meli   V!     S A L D I                 !
!7 Nanni  V! Cognome Fido Saldo etc.pointer !
!8 Pini   P!1 Carli  500                   !
!_________ !2 Dani   700      0        2   !
           !3 Eli    400               5   !
           !4 Fani   300  + 200        1   !
           !5 Giorgi 800
           !6 Meli   250
           !7 Nanni  350
           !8 Pini   600
```

```
-------------------------------------------
!   M O V I M E N T I     puntatori         !
! Causale  Etc.Importi conto last next!
!1 Ns.fattura    +500    4     0    3 !
!2 Ns.fattura    +300    2     0    6 !
!3 Ns.tratta     -100    4     1    4 !
!4 Ns.tratta     -200    4     3      !
!5 Incasso       -100    3     0      !
!6 Ns.tratta     -200    2     2    7 !
!7 Incasso       -100    2     6      !
!-----------------------------------!
```

Figura 7.1

7.5 I modi di gestione dei file

Le operazioni di gestione possono essere registrate in un sistema computerizzato per lotti, alla fine di dati intervalli di tempo, oppure nello stesso momento in cui vengono messe in atto. Si parla, nel primo caso, di registrazione in archivio **Batch-mode**, nel secondo di registrazione **real-time**. Tipicamente vengono registrate in modo Batch le schede di lavoro dei dipendendti e le prime note delle diverse applicazioni.

Se le registrazioni effettuate negli archivi delle transazioni hanno effetto immediato anche negli archivi con queste collegati allora si parla di gestione on-line, ad indicare che tutti gli archivi interessati dalla singola transazione vengono immediatamente registrati.

Una azienda che disponga di più elaboratori in rete, o di un elaboratore con più terminali multitasking, può facilmente effettuare registrazioni di operazioni nello stesso momento in cui le operazioni si realizzano nei diversi archivi cui le operazioni compiute interessano.

Si parla allora di registrazioni Real-time.

CAPITOLO 8

GLI STRUMENTI DI RAPPRESENTAZIONE DEL CONTENUTO DEGLI ARCHIVI

8.1 I tracciati record.
8.2 Le tavole delle variabili.
8.3 Gli indici.

8.1 I tracciati record.

La maggior parte dei programmi per applicazioni gestionali abbisogna di archivi permanenti su cui registrare dati e informazioni necessari alla realizzazione della attività aziendale.

Detti archivi, o file, vanno descritti nelle variabili necessarie per la loro lettura o registrazione, nel loro contenuto e nello spazio occupato.

Essendo un file solitamente composto di record, vanno descritti i record nel loro contenuto, nei campi che li compongono, nel numero dei caratteri occupati.

A tal fine si usano prospetti denominati tracciati record (vedi figura 8.1).

```
                                                  Nome file: ANAG
                        Tracciato record
|-----------------------------------------------------------------------
|  campo  n.1        |campo n.2  |n.3 | campo n.4              |campo
| RAGIONE SOCIALE    | VIA       |CAP | CITTA'                 |
!..../....!..../....!..../....!..../....!..../....!..../....!..../..
-----------------------------------------------------------------------
```

Figura 8.1

8.2 Le tavole delle variabili

I tracciati record sopra descritti sono composti sulla base di tavole di variabili con le quali vengono definiti i nomi delle variabili di campo necessarie alla lettura o registrazione, il tipo delle variabili stesse, il loro contenuto e il numero dei caratteri occupato.

La tavola delle variabili per il record sopradescritto viene descritta nella figura 8.2 :

TAVOLA DELLE VARIABILI DI CAMPO

NOME	TIPO	CONTENUTO	N.CAR.
RA***	ALFABETICO	ragione sociale var. programma	20
RAG***			
VI***	ALFABETICO	Via e var. programma	11
VIA***			
CAP***	ALFABETICO	C.a.p. var.di campo var. programma	4
CAP	NUMERICO SING.		

Figura 8.2

8.3. Gli indici.

Se i record di un file non sono omogenei nella loro struttura, nel loro contenuto e nella loro funzione all'interno del sistema informativo, diventa necessario definire con chiarezza la funzione svolta da ciascuno di essi.

Si provvede allo scopo preparando un indice del file con l'indirizzo di ciascun record e una chiave del suo contenuto e della funzione svolta. Detto indice, che si può rappresentare con una tabella del tipo della fig. di pag. 39, può a sua volta essere registrato in un apposito file.

CAPITOLO 9

LISTATI DELLE SUBROUTINE DI BASE

9.1 Organizzazione di un programma e distribuzione delle righe tra le sue parti.
9.2 Struttura e numeri di riga delle subroutine di base
9.3 Listati delle subroutine di base.

9.1 Organizzazione di un programma e distribuzione delle righe tra le sue parti

E' buona norma predisporre e organizzare una volta per tutte le subroutine di base in modo da distribuire razionalmente le righe di programma tra le diverse applicazioni senza creare sovrapposizioni.

A tal fine si è pensato di distribuire lo spazio per le righe di programma in quattro parti nel modo seguente:

1) da riga 10 a riga 9999 righe riservate alle routine specifiche
2) da riga 10000 a riga 11999 righe riservate alle subroutine di base
3) da riga 12000 a riga 14999 righe riservate alle subroutine di gestione dei file.
4) da riga 15000 in avanti righe riservate per le subroutine di costruzione dei format-video

9.2 Struttura e numeri di linea delle subroutine di base

L'insieme delle subroutine di base occupa le righe da 10000 a 11999
Le righe di programma sono distribuite tra le subroutine nel modo seguente:

1) da riga 10000 a riga 10099 subroutine di DIMENSIONAMENTO
2) da riga 10100 a riga 10195 subroutine di controllo delle DATE
3) da riga 10200 a riga 10270 subroutine di CONVERSIONE DATE
4) da riga 10300 a riga 10430 subroutine di RICONVERSIONE IN DATA
5) da riga 10500 a riga 10520 subroutine di LETTURA FILE
6) da riga 10600 a riga 10620 subroutine di REGISTRAZIONE FILE

7) da riga 10700 a riga 10756 subroutine di CONTROLLO COD. F.
8) da riga 10800 a riga 10960 subroutine di CONTROLLO P. IVA
9) da riga 11000 a riga 11090 subroutine di VISUALIZZAZIONE MASCHERE
10) da riga 11200 a riga 11350 subroutine di CONTROLLI INPUT
11) da riga 11500 a riga 11595 subroutine di ORDINAMENTO INDICE
12) da riga 11600 a riga 11670 subroutine di RICERCA INDICE
13) da riga 11800 a riga 11940 subroutine di STAMPA

9.3 Listati delle subroutine di base (da 9.1 a 9.12).

LISTATO 9.1

```
10000 REM DIMENSIONAMENTO

10010 DIM G%(12),NCAR%(25),TIPO%(25),TMS$(25),
      V%(25),H%(25),TEMP%(25),RISP$(25)
10020 G%(1)=31:G%(2)=28:G%(3)=31:G%(4)=30:G%(5)
      =31:G%(6)=30:G%(7)=31:G%(8)=31:G%(9)=30:
      G%(10)=31:G%(11)=30:G%(12)=31
10030 DIM TIT$(16),T(16),NU(16),DESC$(50),PU%
      (50),TAP$(16)

10090 RETURN
```

LISTATO 9.2

```
10100 REM DATA: variabili in input C, H%(C),
      V%(C), GIORD; variabili in output GIORNO
      , MESE ,ANNO

10130 NCAR%(C)=2:TIPO%(C)=1:H%(C)=H%(C)+2:
      TEMP%(C)=H%(C)

10135 LOCATE V%(C),H%(C),0:PRINT"anno":H%(C)=
      H%(C)+4:GOSUB 11200:H%(C)=H%(C)+4
10140 IF IN$(5)="" AND GIORD>0 THEN RETURN ELSE
      ANNO=VAL(IN$(5))
10145 ANNO=VAL(IN$(5)):IF ANNO>99 OR ANNO<50
      THEN H%(C)=H%(C)-8:GOTO 10135
10150 IF ANNO MOD 4=0 THEN G%(2)=29 ELSE G%(2)
      =28
```

```
10165 LOCATE V%(C),H%(C),0:PRINT"mm":H%(C)=H%(C)
      +2:GOSUB 11200:H%(C)=H%(C)+4
10170 MESE=VAL(IN$(5)):IF MESE>12 OR MESE <0
      THEN H%(C)=H%(C)-6:GOTO 10165

10175 LOCATE V%(C),H%(C),0:PRINT"gg":H%(C)=H%(C)
      +2:GOSUB 11200
10180 GIORNO= VAL(IN$(5))
10185 IF GIORNO<0 OR GIORNO >G%(MESE) THEN H%(C)
      = H%(C)-2:GOTO 10175
10190 H%(C)=TEMP%(C)
10195 RETURN
```

LISTATO 9.3

```
10200 REM CONVERTI DATA: IN INPUT GIORNO, MESE
      ANNO, IN OUTPUT GIORD.

10205 GIORD=0

10210 FOR AN%=50 TO ANNO
10220 IF AN% MOD 4 =0 THEN GIORD=GIORD+366 ELSE
      GIORD=GIORD+365
10230 NEXT AN%

10240 FOR MESD%=1 TO MESE
10250 GIORD =GIORD+G%(MESD%-1)
10260 NEXT MESD%

10270 GIORD=GIORD+GIORNO

10290 RETURN
```

LISTATO 9.4

```
10300 REM CONVERTI ORDINALE IN ANNO, MESE,
      GIORNO.
10310 REM in input GIORD, in output ANNO, MESE,
      GIORNO.

10320 GIORNO=0:MESE=0:ANNO=0:GIAN=365
```

```
10330 WHILE GIORD>GIAN
10340 IF (1900+ANNO)MOD 4= 0 THEN GIAN=366
      ELSE GIAN=365
10350 GIORD=GIORD-GIAN:ANNO=ANNO+1
10360 WEND

10370 ANNO=49+ANNO

10380 WHILE GIORD>G%(MESE+1)
10390 GIORD=GIORD-G%(MESE+1):MESE=MESE+1
10400 WEND


10410 MESE=MESE+1
10420 GIORNO=GIORD
10425 DDTA=ANNO*10000+MESE*100+GIORNO

10430 RETURN
```

LISTATO 9.5

```
10500 REM LETTURA
10510 GET #NF%,R%
10520 RETURN
```

LISTATO 9.6

```
10600 REM SCRITTURA
10610 PUT #NF%,R%
10620 RETURN
```

LISTATO 9.7

```
10700 REM CONTROLLO DEL CODICE FISCALE.
      VARIABILI IN INPUT CODF$,IN OUTPUT FLAGC

10702 DIM CD(26),C(16):RESTORE

10704 FOR I= 1 TO 16: READ C(I):NEXT I:
      DATA 0,0,0,0,0,0,1,1,0,1,1,0,1,1,1,0
```

```
10706 FOR I = 1 TO 26:READ CD(I):NEXT I:
      DATA 1,0,5,7,9,13,15,17,19,21,2,4,18,20,
      11,3,6,8,12,14,16,10,22,25,24,23
10708 FLAGC=1:PRINT CODF$
10710 IF LEN(CODF$)<>16 THEN 10756

10712 FOR I=1 TO 15
10714 CH$=MID$(CODF$,I,1)
10716 IF(CH$>="A" AND CH$<="Z")AND C(I)<>0 THEN 10756
10718 IF(CH$>="0"AND CH$<="9")AND C(I)<>1 THEN 10756
10720 NEXT I
10722 S=0

10724 FOR I=1 TO 15 STEP 2
10726 CH$=MID$(CODF$,I,1)
10728 IF C(I)=0 THEN 10734
10730 S=S+CD(VAL(CH$)+1)
10732 GOTO 10736
10734 S=S+CD(ASC(CH$)-64)
10736 IF I=15 THEN 10748
10738 CH$=MID$(CODF$,I+1,1)
10740 IF C(I+1)=0 THEN 10746
10742 S=S+VAL(CH$)
10744 GOTO 10748
10746 S=S+ASC(CH$)-65
10748 NEXT I

10750 R=S-INT(S/26)*26
10752 IF MID$(CODF$,16,1)<> CHR$(R+65) THEN 10756
10754 FLAGC=0

10756 RETURN
```

LISTATO 9.8

```
10800 REM CONTROLLO PARTITA IVA: IN INPUT
      PIVA$, IN OUTPUT FLIVA. FLIVA E' 0 SE
      IL NUMERO E' CORRETTO 1 SE ERRATO.

10820 S=0:FLIVA=1

10830 IF VAL(LEFT$(PIVA$,7))=0 THEN 10960'
      CONTROLLO DELLE PRIME SETTE CIFRE CHE NON
      DEBBONO ESSERE NULLE
```

```
10840 OD=VAL(MID$(PIVA$,8,3)):IF OD<1 OR OD>95
      THEN 10960' LE CIFRE TRA 8 E 11 DEBBONO

10850 FOR I = 1 TO 10 STEP 2

10860 CIF$=MID$(PIVA$,I,1):IF CIF$<"0" OR CIF$>
      "9" THEN 10960
10870 S=S+VAL(CIF$)'        SOMMA CIFRA DI POSTO
                                          DISPARI

10880 CIF$=MID$(PIVA$,I+1,1):IF CIF$<"0" OR
      CIF$>"9" THEN 10960
10890 CIF=VAL(CIF$)
10900 S=S+2*CIF-9*INT(CIF/5)

10910 NEXT I

10920 T=10-(S-INT(S/10)*10)'   COMPLEMENTO A 10
10930 IF T=10 THEN T=0
10940 IF T<> VAL(RIGHT$(PIVA$,1)) THEN 10960
10950 FLIVA=0

10960 RETURN
```

LISTATO 9.9

```
11000 REM VISUALIZZAZIONE MASCHERA: in input
      NINPUT%,V%(C),H%(C),TMS$(C),in output la
      videata e H%(C)

11010 CLS' Cancella lo schermo

11020 FOR C=1 TO NINPUT%

11030 LOCATE V%(C),H%(C),0'
                              visualizza le scritte
11040 PRINT TMS$(C)'

              calcola le posizioni degli input
11050 H%(C)=H%(C)+LEN(TMS$(C))+2

11060 NEXT C

11070 RETURN
```

LISTATO 9.10

```
11200 REM CONTROLLO  DELL' INPUT SU VIDEO: in
      input V%(C),H%(C),NCAR%(C),TIPO%(C), in
      output IN$(4), IN$(5).

11205 LOCATE V%(C),H%(C)+1,1
11210 FOR J=1 TO NCAR%(C):PRINT"-";:NEXT J:PRINT
      TAB(H%(C))" ";
11220 IN$(5)="":IN$(4)="":J1=1

11230 FOR J=J1 TO NCAR%(C)

11250 IF TIPO%(C)=1 THEN LOCATE V%(C),H%(C)+
      NCAR%(C)-J,1 ELSE LOCATE V%(C),H%(C)+J,1
11255 PRINT " ";:IN$(4)=INPUT$(1)
11260 IF IN$(4)=CHR$(13) THEN RETURN
11270 IF IN$(4)=CHR$(8) AND J>=2 THEN J=J-2:
      IN$(5)=LEFT$(IN$(5),J):IN$(4)="":
      GOTO 11290
11275 IF IN$(4)=CHR$(8) AND J=<1 THEN BEEP:
      GOTO 11230
11280 IF TIPO%(C)=1 AND ((IN$(4)<"0")OR(IN$(4)>
      "9")) THEN 11250

11285 IN$(5)=IN$(5)+IN$(4)
11290 IF TIPO%(C)=1 THEN LOCATE V%(C),H%(C)+
      NCAR%(C)-J+1,1:PRINT IN$(5) ELSE LOCATE
      V%(C),H%(C)+J,1:PRINT IN$(4)

11300 NEXT J

11310 LOCATE 23,67,1:PRINT"O K";:IN$(4)=
      INPUT$(1):IF IN$(4)=CHR$(8) THEN J1=J-2:
      GOTO 11230:ELSE IF IN$(4)="N" THEN 11205

11350 RETURN
```

LISTATO 9.11

```
11500 REM RIORDINA INDICE - Bubble Sort -
      paragrafo 7.1 dell'appendice.


11505 REM Variabili interne NF%,R%,N%,
come n. logico di file ,numero record e n. record
riservati all'indice. DESC$(I) e PU%(I) sono le
chiavi e i puntatori da ordinare.
```

```
  Legge il record di servizio dell'indice
11510 NF%=1:R%=1:GOSUB 10500:GOSUB 14700:N%=ID%
      :ID%=0:NRI%=0

  Legge i record dell'indice vero e proprio
11520 FOR RIG%=2 TO N%
11530 R%=RIG%:GOSUB 10500:GOSUB 14700:DESC$(RIG%)
      =DESC$:PU%(RIG%)=PU%
11540 NEXT RIG%

11550 FLAG=0
11555 FOR I%=2 TO N%
11560 IF DESC$(I%)>DESC$(I%+1)AND DESC$(I%+1)
 <>" " THEN TEMP$=DESC$(I%):DESC$(I%)= DESC$
 (I%+1): DESC$(I%+1)=TEMP$:TEMP%=PU%(I%):PU%
 (I%)=PU%(I%+1):PU%(I%+1)=TEMP%:FLAG=1
11565 NEXT I%

11570 IF FLAG =1 THEN 11550

11575 FOR I%=2 TO N%
11580 R%=I%:DESC$=DESC$(I%):PU%=PU%(I%):GOSUB
      14600:GOSUB 10600
11590 NEXT I%

11595 RETURN
```

LISTATO 9.12

```
11600 REM RICERCA INDICE [IN INGRESSO CH$,ID%
      IN USCITA FL,CH E R%]

11610 LI=2:LS=ID%:LUN=LEN(CH$)
11620 IF LS<LI THEN FL=0:CH=0:RETURN:REM NON
      TROVATO
11630 I=INT((LI+LS)/2):R%=I:GOSUB 10500:GOSUB
      14700
11640 IF CH$<LEFT$(DESC$,LUN) THEN LS=I-1:
      GOTO 11620
11650 IF CH$>LEFT$(DESC$,LUN) THEN LI=I+1:
      GOTO 11620
11660 FL=1:CH=I:REM TROVATO

11670 RETURN
```

Vedasi il paragrafo 7.2.3 dell'appendice M 6

CAPITOLO 10

GLI OBBLIGHI CONTABILI

10.1 I soggetti obbligati alla tenuta delle scritture contabili.
10.2 I regimi di impresa:
10.2.1 il regime ordinario;
10.2.2 il regime semplificato;
10.2.3 il regime speciale per l'agricoltura e la pesca.
10.3 Gli obblighi di tenuta e conservazione delle scritture contabili:
10.3.1 il GIORNALE

La progettazione di procedura di contabilità richiede in primo luogo una analisi del problema nei suoi aspetti giuridici, ragioneristici e organizzativi. Si ritiene pertanto opportuno premettere alle considerazioni sulla realizzazione delle procedure le informazioni sugli obblighi giuridici principali riguardanti la tenuta delle scritture contabili.

10.1 Soggetti obbligati alla tenuta delle scritture contabili (art. 13 D.P.R. 29/9/1973 n. 600)

Sono obbligati per detto articolo 13 alla tenuta delle scritture contabili i seguenti soggetti:

a) le società soggette all'imposta sui redditi delle persone giuridiche;

b) gli enti pubblici e privati diversi dalle società soggetti alla imposta sul reddito delle persone giuridiche che hanno per oggetto l'esercizio di attività commerciali;

c) le società soggette all'imposta sul reddito delle persone fisiche

d) le persone fisiche che esercitano imprese commerciali ai sensi del l'art. 51 del D.P.R. 29/9/1973 n.597;

e) le persone fisiche che esercitano arti o professioni ai sensi del l'art. 49 del D.P.R. 29/9/1973 n.597;

f) le società costituite da artisti e le società di professionisti (art.5 D.P.R. 29/9/1973 n.597).

10.2 I regimi di impresa (art. 14 D.P.R. 29/9/1973 n.600)

Gli obblighi contabili delle imprese commerciali sono differenziati in relazione alla categoria di volume d'affari delle imprese stesse.
Si distinguono pertanto le seguenti categorie:

- a REGIME ORDINARIO, con volume d'affari superiore a 780 milioni;

- a CONTABILITA' SEMPLIFICATA per le imprese minori con volume d'affari fino a 780 milioni;

- a REGIME SPECIALE per pesca e agricoltura.

10.2.1 Il regime ordinario.

Le scritture contabili delle imprese commerciali, delle società e degli enti equiparati sono le seguenti:

1) LIBRO GIORNALE (art. 14 D.P.R. 29/9/73 n.600);

2) LIBRO DEGLI INVENTARI (artt. 14-15);

3) REGISTRO DEI BENI AMMORTIZZABILI (art.16);

4) REGISTRO DEI CODICI delle contabilità automatizzate (art. 14);

5) SCRITTURE AUSILIARIE (conti di mastro);

6) SCRITTURE DI MAGAZZINO (D.P.R. 4/11/1981 N.664);

7) REGISTRI I.V.A. (vedi D.P.R 26/10/1972 n.633).

10.2.2 Il regime semplificato

I registri obbligatori per le imprese minori che non abbiano superato un volume d'affari di 780 milioni nell'anno 1984 sono i seguenti:

1) REGISTRO IVA ACQUISTI

2) REGISTRO DEGLI INVENTARI

3) REGISTRO IVA DELLE FATTURE EMESSE e o dei corrispettivi.

Nel registro degli acquisti si registrano anche le operazioni non IVA e si possono annotare le rimanenze di inventario di fine anno entro il termine per la dichiarazione dei redditi.

10.2.3 Il regime speciale per l'agricoltura (art. 34 D.P.R. 633/1972).

Il regime speciale per l'agricoltura esonera dagli adempimenti IVA con l'eccezione delle fatture di acquisto gli agricoltori con volume d'affari inferiore a 10 milioni.

Gli agricoltori con volume d'affari superiore a 10 milioni debbono adempiere ai seguenti obblighi:

1) conservazione e numerazione delle fatture di vendita;

2) conservazione e numerazione delle fatture di acquisto;

3) registrazione degli acquisti e delle vendite.

(1)

10.3 Gli obblighi di tenuta e conservazione delle scritture contabili.

L'articolo 22 del D.P.R. 29/9/1973 n. 600 stabilisce che debbono essere numerate e bollate secondo l'articolo 2215 del codice civile le seguenti scritture:

a) il libro giornale;
b) il libro degli inventari;
c) il registro dei beni ammortizzabili;
d) le scritture ausiliarie di magazzino;
e) i registri IVA;
f) il registro dei codici delle scritture automatizzate:
g) i registri della contabilità semplificata;
h) i registri degli esercenti arti e professioni.

Per detto articolo 2215 si prescrive che i libri di cui sopra, prima di essere messi in uso, debbano essere numerati progressivamente in ogni pagina e bollati in ogni foglio e debbano avere nell'ultima pagina la indicazione del numero dei fogli di cui il registro si compone.

(1) Il decreto 19/12/1984 n. 853 fa obbligo alle imprese commerciali che non hanno conseguito ricavi per un ammontare superiore a 780 milioni nell'anno 1984 di optare tra il regime ordinario e un regime forfettario sia ai fini IVA che a quelli delle imposte sui redditi.

Le scritture cronologiche, come il libro giornale ad esempio, debbono essere eseguite entro il termine di 60 giorni dalla data di ricezione o emissione del documento che rileva il fatto amministrativo da registrare.

Deve essere tenuto in considerazione anche l'art. 2219 del codice civile che prescrive che tutte le scritture siano tenute secondo le norme di una ordinata contabilità, senza spazi in bianco, senza interlinee e senza trasporti in margine.

Si ricorda ancora che il GIORNALE e il LIBRO DEGLI INVENTARI devono essere annualmente vidimati dall'ufficio del registro delle imprese o da un notaio.

Si nota infine che le scritture ausiliarie tipo LIBRO MASTRO, REGISTRI DI MAGAZZINO, CONTI INDIVIDUALI possono essere fatte senza formalità specifiche.

Tutte le scritture contabili, compresi i supporti elettronici, devono essere conservati, fino a quando non siano stati definiti gli accertamenti relativi al corrispondente periodo di imposta, eventualmente anche oltre il termine di 10 anni stabilito dall'articolo 2220 del codice civile o da altre leggi tributarie

10.3.1 Il GIORNALE.

Il libro GIORNALE prima di essere attivato deve essere numerato progressivamente in ogni pagina e bollato in ogni foglio (art.2215); deve contenere l'indicazione cronologica delle operazioni relative all'esercizio dell'impresa ed essere annualmente vidimato.
Il registro di PRIMANOTA, regolarmente bollato e vidimato prima dell'uso, contenente in ordine cronologico tutti i dati necessari alla identificazione delle operazioni di impresa può essere considerato un vero e proprio giornale con piena validità giuridica e fiscale.

Ai sensi del comma 1 dell'art. 22 del D.P.R. 29/9/1973 n.600 si nota che "le registrazioni cronologiche devono essere eseguite non oltre i 60 giorni" dalla data di ricezione o emissione del documento che prova la effettuazione della operazione da registrare. Tale termine è stato unificato anche per le registrazioni I.V.A. effettuato con macchine elettroniche o elettrocontabili (D.M. 11/8/75).

Nel caso in cui si impieghino codici per le denominazioni dei conti e per le causali delle operazioni da registrare è opportuno riportare in chiaro le causali stesse e in ogni caso si deve predisporre un registro dei codici e delle causali utilizzati con le relative spiegazioni.

Nel giornale possono essere riepilogate quotidianamente le operazioni soggette ad IVA purchè dette operazioni risultino da registri IVA preventivamente bollati e vidimati e analiticamente redatti a termini di legge.

CAPITOLO 11

L'AZIENDA COME OGGETTO DI OSSERVAZIONE E RILEVAZIONE

11.1 L'azienda e le operazioni della sua gestione.
11.2 La nozione di reddito globale e di reddito di esercizio.
11.3 La composizione del reddito di esercizio.
11.4 Il metodo della partita doppia.
11.5 La serie dei conti che consentono la duplice osservazione e rilevazione degli scambi.
11.6 Il piano dei conti.

11.1 L'azienda e le operazioni della sua gestione

Una azienda è strumento per lo svolgimento sistematico di attività economica per il consumo,la produzione e lo scambio di mercato.

Le aziende che svolgono attività economica per il consumo di erogazione si dicono aziende di consumo, quelle che consumano per la produzione e producono per lo scambio si dicono imprese.

Nelle imprese l'attività economica consiste di operazioni che hanno per oggetto scambi in acquisto, trasformazioni tecnico-economiche, scambi in vendita di beni e di servizi. Si tratta di operazioni che hanno per oggetto l'acquisto di beni economici regolate in moneta o in crediti di moneta (vedi fig. 11.1).

```
 Scambi in acquisto                          Scambi in vendita

            USCITE                            ENTRATE
<-------moneta--<     Trasformazioni            <-----moneta--<

          ACQUISTI   tecnico- econo          VENDITE
>- beni -------->          miche                >- beni ----->
```

Figura 11.1

11.2 La nozione di reddito globale e di reddito di esercizio.

La valutazione dei risultati delle aziende implica la valutazione di sintesi della sua attività (della sua gestione) e delle condizioni in cui si è operato.

Le operazioni costituiscono una unità: in effetti sono legate da vincoli sia nella simultaneità che nella successione del tempo.

Data la unità delle operazioni di gestione una prima nozione di risultato economico può fare riferimento a tutta la attività realizzata dalla impresa nella sua vita. Se si considera la gestione nella sua globalità si determina quello che prende il nome di reddito globale.

Se una impresa è stata costituita con un dato sistema di beni capitali, impianti, macchine, denaro, il risultato economico alla fine della sua vita è dato dalla differenza tra il valore iniziale e il valore finale del capitale.

Tutto ciò è vero se non vi sono stati apporti o prelievi di beni capitali e se non è cambiato il valore della moneta utilizzato come strumento di rappresentazione del valore del capitale.

Lo stesso risultato può essere ottenuto calcolando la differenza tra tutte le entrate e tutte le uscite di moneta o di crediti dovute agli scambi per le operazioni di gestione.

Pertanto possiamo dire che :

1) reddito globale = capitale finale - capitale iniziale

2) reddito globale = entrate - uscite

Si nota che il risultato economico globale non risponde a concrete esigenze amministrative. Anche se le operazioni compiute costituiscono un tutto unitario vi sono esigenze concrete che impongono di determinare periodicamente il reddito di impresa.

Si tratta innanzitutto della esigenza di stabilire in quale misura sia possibile effettuare dei prelevamenti di mezzi dalla azienda a titolo di remunerazione del capitale dei soci, dell'imprenditore, degli amministratori, senza pregiudicarne la continuità; si tratta di avere indicazioni sulla futura attività dell'impresa e sulla sua convenienza.

E' necessario allora introdurre delle :potesi di scindibilità della gestione ed individuare delle unità economiche relative di operazioni, compiute in periodi della durata di un anno, perchè è alla fine di periodi di questa durata che normalmente di ripartiscono gli utili ai soci.

Le unità relative di operazioni di gestione prendono il nome di esercizi;
i periodi di tempo in cui quelle operazioni si svolgono prendono il nome di periodi amministrativi.

La individuazione delle relazioni che legano le operazioni in esercizio, la determinazione della competenza, non si fonda tanto sulla appartenenza temporale delle loro entrate e delle loro uscite al periodo amministrativo di cui si deve determinare il reddito, ma sul concorso (la competenza) di dette operazioni alla produzione dei redditi in questione.

Si osserva che beni e servizi il cui acquisto ha dato luogo ad uscite di moneta o a debiti in un dato periodo amministrativo di fatto concorrono alla produzione dei redditi di periodi amministrativi diversi.

Diventa necessario considerare gli scambi anche in un aspetto diverso da quello, pur importante, delle entrate e delle uscite di moneta detto aspetto monetario. Diventa necessario considerare gli scambi nel loro aspetto economico che evidenzia le caratteristiche economiche dei beni e dei servizi acquisiti e consente di meglio definire il tipo del concorso di detti beni alla produzione dei redditi che si vogliono determinare.

11.3 La composizione del reddito di esercizio

Sia detto ricavo il valore dei beni o dei servizi ceduti in uno scambio.

Sia detto costo il valore dei beni o dei servizi acquisiti.

Si dicano accertati quei costi e quei ricavi che hanno dato luogo a delle entrate o a delle uscite di moneta, debiti o crediti.

Se è vero che il reddito di esercizio deve essere determinato per differenza tra ricavi e costi di competenza per determinare detta competenza si deve procedere con le operazioni seguenti:

1) somma algebrica dei ricavi e dei costi accertati nel periodo amministrativo:

2) rettifiche sottrattive di valori di costi e di ricavi di competenza dei futuri esercizi;

3) rettifiche aggiuntive di ricavi e di costi, o di loro parti, costi e ricavi che, pure accertati in altri esercizi, si ritengano di competenza dell'esercizio di cui si deve determinare il reddito.

Le rettifiche sottrattive di costi e di ricavi si fondano sulla valutazione dei beni che costituiscono il capitale alla fine del periodo amministrativo. Si noti che la valutazione del capitale costituisce una condizione del calcolo del reddito di esercizio.

Data la procedura di calcolo possiamo dire che il reddito di esercizio è costituito di componenti positivi e negativi come nel seguente prospetto:

```
Ricavi accertati nel periodo amministrativo   +
rettifiche aggiuntive di ricavi di competenza +
rettifiche sottrattive di costi               =
---------------------------------------------

                          COMPONENTI  POSITIVI     -
Costi accertati nel periodo amministrativo    +
Rettifiche aggiuntive di costi di competenza  +
Rettifiche sottrattive di ricavi              =
---------------------------------------------
                          COMPONENTI NEGATIVI     =
                          -----------------------
                          REDDITO DI ESERCIZIO
```

Figura 11.2

11.4 Il metodo della partita doppia.

E' riconosciuta la utilità della osservazione delle operazioni per le seguenti finalità:

- **controllo dei movimenti numerari**; (controllo dei movimenti delle entrate e delle uscite di cassa e della sua giacenza, dei crediti concessi, riscossi, da riscuotere, dei debiti ottenuti e da pagare)

- **determinazione del reddito di esercizio**.

Le due finalità implicano la considerazione degli scambi monetari nei due aspetti seguenti:

1) **aspetto monetario** inerente il regolamento del prezzo, il pagamento in contanti o differito sulla parola o con cambiali etc.;

2) **aspetto economico** inerente le caratteristiche economiche dei beni e dei servizi acquisiti o ceduti negli scambi per il loro concorso alle produzioni dei redditi.

Si deduce la possibilità di una simultanea abbinata rilevazione dei valori che lo scambio origina perchè la moneta da pagare o da riscuotere nello scambio è espressione di entrambi gli aspetti da considerare.

E infatti:

a) la determinazione e il controllo dei movimenti numerari implica la rilevazione della quantità di moneta pagata o da pagare, incassata o da incassare negli scambi;

b) la determinazione del reddito di esercizio richiede da un lato la rilevazione dei beni e dei servizi acquisiti nella grandezza derivata dalla quantità di moneta pagata o da pagare e dei servizi e dei beni ceduti nella grandezza derivata dalla quantità di moneta per lo scambio incassata o da incassare.

Il metodo che consente detta duplice rilevazione simultanea dei fatti amministrativi prende il nome di metodo della partita doppia.

Detto medoto si serve di due serie di conti (vedasi pag. 68) destinata ognuna a rilevare valori relativi ad uno degli aspetti tipici di osservazione degli scambi.

I valori rilevati complessivamente in ciascuna serie sono uguali, uguali essendo i valori di ogni singolo scambio rilevati nei conti delle due serie contrapposte. Tale uguaglianza implica il principio dello antitetico funzionamento dei conti appartenenti a serie diverse.

Scelti i conti destinati ad accogliere i valori di un dato scambio , la rilevazione con un dato segno algebrico del relativo credito di regolamento implica la rilevazione, con segno opposto, del valore della vendita della merce corrispondente.

Da ciò discende che, salvo errori, il totale dei valori registrati con un dato segno algebrico è uguale al totale dei valori registrati con segno opposto.

Tale metodo richiede, oltre agli strumenti di classificazione dei valori, anche di uno strumento che documenti la successione temporale dei fatti amministrativi.

La registrazione nei conti dei valori attinenti fatti amministrativi accertati da documenti di prova o di autorizzazione dovrà essere preceduta da rilevazioni in giornale.

11.5 Le serie di conti che consentono la duplice osservazione e rilevazione degli scambi

L'ordinare le singole manifestazioni di un fenomeno complesso, o di fenomeni diversi appartenenti alla stessa classe, secondo gli attributi che essi presentano è comporre una serie. Le serie vengono di solito registrate in tabelle che ne sono strumento di rappresentazione e di successiva elaborazione. Tra le tabelle molto importanti sono i conti, strumenti atti ad accogliere serie di valori di segno opposto.

Le quantità di segno opposto relative a variazioni negative o positive dell'oggetto di un conto si dicono variazioni di conto.

I conti sono spesso divisi in due sezioni, una per una serie di valori, l'altra per la serie di segno contrario. Si conviene di attribuire alla serie di sinistra il segno DARE, a quella di destra il segno AVERE. Se il conto è intestato ad una persona un valore in DARE indica che quella persona deve DARE quel valore mentre un valore in AVERE indica che quella persona deve AVERE quel valore.

ACCENDERE un conto significa iscrivere nello stesso la prima variazione.

ADDEBITARE un conto significa rilevare nel conto un valore in DARE.

ACCREDITARE un conto significa rilevare una variazione in AVERE.

Per MOVIMENTO DARE si intende il TOTALE delle variazioni in DARE

Per MOVIMENTO AVERE si intende il TOTALE delle variazioni in AVERE.

Per SALDO si intende la differenza tra il MOVIMENTO DARE e quello AVERE.

La osservazione degli scambi nei due aspetti economico e numerario per le descritte esigenze di determinazione del reddito e di controllo dei movimenti numerari suscita la esigenza di raggruppare i conti in due serie distinte "differenziate nel significato dei valori che accolgono e rese antitetiche nelle modalità di funzionamento."

Le due serie di conti prendono i seguenti nomi:

a) CONTI NUMERARI

b) CONTI ECONOMICI.

Per quanto riguarda il funzionamento delle due serie di conti è stabilito convenzionalmente quanto segue:

a) i CONTI NUMERARI accolgono variazioni numerarie attive in DARE e variazioni numerarie passive in AVERE.

b) i CONTI ECONOMICI accolgono variazioni economiche positive in AVERE e variazioni negative in DARE.

Tra i CONTI ECONOMICI i conti che rilevano riassuntivamente il capitale della impresa e i suoi componenti, mettendone in evidenza la particolare fonte di raccolta di mezzi di finanziamento e i vincoli che ne derivano nei confronti del capitale raccolto da terzi, accolgono in AVERE variazioni positive del capitale proprio e in DARE variazioni negative.

I conti nella loro forma tradizionale presentano il seguente contenuto:

```
                    CONTI AI VALORI NUMERARI
          ATTIVI                              PASSIVI
----------------+----------------   ----------------+----------------

Dare            |          Avere   Dare            |          Avere
                |                                  |
CONSISTENZA INI |                                  |CONSISTENZA INI
ZIALE           |                                  |ZIALE
                |                                  |
VARIAZIONI IN + |VARIAZIONI IN -  VARIAZIONI IN -  |VARIAZIONI IN +
DI VALORI ATTIVI|                                  |DI VALORI PASSIVI
                |                                  |
                |(SALDO)                 (SALDO)   |


                        CONTI ECONOMICI
             COSTI                                RICAVI
----------------+----------------  ----------------+----------------
Dare            |          Avere  Dare            |          Avere
                |                                 |
COSTI           |                                 |          RICAVI
                | (SALDO)                (SALDO)  |
```

Figura 11.3

11.6 Il piano dei conti

La rilevazione delle operazioni di gestione richiede l'impiego di un sistema di conti il cui funzionamento e i cui vincoli devono essere accuratamente previsti prima della loro effettiva utilizzazione; si deve comporre un PIANO di CONTI atto alla rilevazione della futura gestione.

In detto piano debbono essere descritti tutti i conti che si prevede di dover utilizzare. Ogni conto deve essere illustrato nei seguenti aspetti:

- il contenuto;
- il momento della rilevazione;
- i collegamenti;
- la codifica.

Di ogni conto deve essere descritto il contenuto. Si deve stabilire, ad esempio, se il costo di acquisto da iscrivere debba essere al netto o al lordo delle spese di facchinaggio, trasporto etc.. Si deve stabilire se nei conti fornitori e clienti si debbano registrare tutte le fatture o solo quelle relative a regolamenti differiti etc..

Di ogni conto deve essere definito il momento della rilevazione. Detto momento di solito corrisponde al momento in cui la azienda emette un documento di prova o di autorizzazione dell'operazione, o ne viene in possesso.
Devono essere indicati i documenti originari dai quali si devono attingere i dati da rilevare, le note di accredito o addebito, le fatture, le reversali di incasso e gli ordini di pagamento etc..

Si indicano anche i giornali, i libri, gli scadenzari, i partitari nei quali debbono essere rilevate le operazioni compiute nei loro aspetti elementari. In tal modo si riallaccia il piano dei conti a tutto il sistema delle rilevazioni contabili che si ritiene di realizzare per la rappresentazione della gestione.

Si devono indicare i collegamenti tra i diversi conti e stabilire il diverso grado di analisi di ciascuno di essi. In rapporto a ciò sorge l'esigenza di definire i collegamenti tra conti sintetici o di gruppo e i relativi conti di mastro e tra questi e i conti di partitario.

I conti debbono essere facilmente identificabili sia individualmente che nelle loro categorie di appartenenza. A tal fine si seguono procedimenti di classificazione e numerazione che prendono il nome di procedimenti di codifica.
Si possono tra gli altri avere i seguenti procedimenti:

- codifica numerica progressiva;
- numerica decimale;
- numerica alfabetica.

Il procedimento di classificazione numerico progressivo permette due soli livelli di osservazione. Si attribuiscono numeri progressivi ai diversi conti di sintesi quindi, dopo un punto di separazione si attribuiscono numeri progressivi ai diversi conti di analisi. Ad esempio, il conto 4.13 sta ad indicare il tredicesimo conto di partitario del quarto conto di mastro.

Il procedimento di classificazione numerico decimale consente diversi livelli di osservazione e rilevazione della gestione. Dai conti di sintesi a quelli di analisi, per ogni livello di osservazione, si procede con la attribuzione di un numero dallo zero al nove. Ad esempio il conto 103 sta ad indicare il conto 1 (Conti numerari), il conto di mastro 10 (Erario con IVA), il conto di analisi 103 (IVA per RETTIFICHE di ACQUISTO).

Si può utilizzare ancora, assieme alla classificazione numerico-decimale anche la classificazione alfabetica. Così i conti di CLIENTI EMILIA, 010, possono essere disposti in ordine alfabetico con al primo posto il cliente ANSELMI e all'ultimo ZECCHI.

Nota
Il consiglio nazionale dei ragionieri e periti commerciali ha elaborato un "piano generale dei conti". Detto piano è articolato in quattro suddivisioni: CLASSI, GRUPPI, CONTI, SOTTOCONTI.
Le classi prescelte per la strutturazione del piano sono: 1. Fonti di finanziamento durevole - 2. Immbilizzazioni - 3. Scorte - 4. Crediti e debiti correnti -5.Liquidità - 6. Costi - 7. Ricavi - 8. Conti di risultato - 9. Conti d'ordine o contabilità analitica.

CAPITOLO 12

CONTABILITA' GENERALE: GLI OUTPUT RICHIESTI

12.1 Il libro Giornale;
12.2 I mastri di conto;
12.3 I registri IVA:
12.3.1 il registro delle Fatture di vendita;
12.3.2 il registro delle Fatture di acquisto;
12.3.3 il registro dei Corrispettivi;
12.3.4 gli allegati Clienti e Fornitori.
12.4 Sintesi dei dati richiesti in OUTPUT.
12.5 Analisi dei dati richiesti in OUTPUT:
12.5.1 i dati simili ripetuti;
12.5.2 i dati costanti e i dati temporanei.

12.1 Il libro Giornale

Il Giornale bollato viene stampato con forma diversa da quella abitualmente utilizzata in una contabilità tradizionale. In ogni caso le stampe devono essere comprensibili e facilmente riconducibili agli schemi tradizionali. Nel caso si impieghino codici per le denominazioni dei conti e per le causali delle operazioni è opportuno riportarne in chiaro il significato.
La stampa del giornale deve riguardare i seguenti elementi essenziali:

- data di registrazione;
- numero di registrazione;
- data documento;
- numero documento;
- denominazione conto;
- descrizione dell'operazione;
- importo DARE;
- importo AVERE.

La forma del giornale dovrebbe essere la seguente:

GIORNALE GENERALE

Data reg.	Num. N.reg	Data doc.	Num. doc.	Descrizione	Conto	Importi DARE	Importi AVERE
10/10	750	9/10	450	Ns.fattura	Clienti	944.000	
10/10	750	9/10	450	Ns.fattura	Vendite A		800.000
10/10	750	9/10	450	Ns.fattura	IVA V 18		144.000
10/10	751	9/10	451	incasso fatt	Cassa	944.000	
10/10	751	9/10	451	incasso fatt.	Clienti		944.000
10/10	752	1/10	833	Vs fattura	AcquistiB	650.000	
10/10	752	1/10	833	Vs fattura	IVA AC 18	117.000	
10/10	752	1/10	833	Vs fattura	Fornitori		767.000
Totale movimenti del 10/10						2655.000	2655.000

Figura 12.1

12.2 I mastri di conto

Per ciascun conto o sottoconto la procedura deve essere in grado di fornire tutti i movimenti di un dato periodo di osservazione nel loro ordine cronologico eventualmente con il conto di contropartita.

Conto Clienti

Data reg.	Num. N.reg	Data doc.	Num. doc.	Descrizione	Conto di	Importi DARE	Importi AVERE
10/10	750	9/10	450	Ns.fattura	Vendite A	944.000	
10/10	751	9/10	451	incasso fatt.	Cassa		944.000
Progressivi attuali						944.000	944.000
Saldo						0	0

Figura 12.2

Per ciascun Cliente è opportuno disporre dei seguenti dati utilizzabili in occasioni diverse durante il periodo amministrativo e a fine anno anche per obblighi imposti dalla disposizione IVA:

- Codice Cliente	- Categoria	- Saldo
- Nome	- Codice agente	- Imponibile
- Indirizzo	- Codice zona	- Non imponibile
- Citta'	- Banca d'appoggio	- Esente
- C.A.P.	- Agenzia	- IVA
- Partita IVA	- Insoluti	- N. fatture
	- Resi	

Figura 12.3

12.3 I registri IVA

I registri che debbono essere tenuti da una azienda in regime ordinario sono i seguenti:

registro Vendite;
registro Acquisti;
registro Corrispettivi;
gli allegati Clienti e Fornitori.

12.3.1
Il registro delle vendite può essere tenuto nella seguente forma:

REGISTRO DELLE FATTURE DI VENDITA

Registrazione data	Documento num	data	num.	Nominativo	Note	Imponibile	IVA %	Importo
10/10	750	9/10	450	Clienti X	ns.fat	800.000	18	144.000
10/10	760	9/10	451	Romei	ns fat	450.000	14	63.000
10/10	763	9/10	452	Motor d	Ind 19	2.000.000	18	360.000

Figura 12.4

12.3.2
Il registro delle fatture di acquisto può essere tenuto in forma simile a quella impiegata per il registro delle vendite:

REGISTRO DELLE FATTURE DI ACQUISTO

Registrazione data	num	Documento data	num.	Nominativo	Note	Imponibile	IVA %	Importo
10/10	752	1/10	833	Ponchi spa	Vs.fat	650 000	18	767.000

Figura 12.5

12.3.3

Il registro dei corrispettivi può essere tenuto nella seguente forma:

REGISTRO DEI CORRISPETTIVI

Registrazione num.	data	Importo Corrispettivi	Assoggettame. Importo	%IVA	Variazioni Importi	%IVA	n.s.,n.i,es Importi	%IVA

Figura 12.6

12.3.4

Gli elenchi Clienti e Fornitori possono essere tenuti nella forma seguente:

ELENCO CLIENTI

Num	Rag. sociale	Cod.fisc	Tot. Imponib.	Tot. IVA Addebiti	Tot. IVA NS NI ES	Tot. Fatture	Num fatt
1	Alpha Mi Via Roma 3 Reggio E		20000000	3600000		23600000	7

Figura 12.7

12.4 Sintesi dei dati richiesti in OUTPUT

In sintesi i dati output della contabilità risultano dagli schemi seguenti:

```
Giornale
-----------------------------------------------------------------------
|Data | Num.|Data | Num.| Descrizione  | Conto     |    Importi      |
| reg.|N.reg| doc.| doc.|              |           | DARE   | AVERE  |
|-----|-----|-----|-----|--------------|-----------|--------|--------|

Estratto conto
-----------------------------------------------------------------------
|Data | Num.|Data | Num.| Descrizione  | Conto di  |    Importi      |
| reg.|N.reg| doc.| doc.|              |           | DARE   | AVERE  |
|-----|-----|-----|-----|--------------|-----------|--------|--------|

 Conto
-------------------------------------------
|C/|  |                  |        |        |
|G |  | Denominazione    |Totale  |Totale  |
|r |T |                  |        |        |
|u |i |        conto     |  DARE  | AVERE  |
|p |p |                  |        |        |
|p |o |                  |        |        |
|o |  |                  |        |        |
|--|--|------------------|--------|--------|

Registro fatture
-----------------------------------------------------------------------
|Registrazione|Documento | Nominativo| Note |Imponibile|IVA|Importo |
|data   | num |data |num.|           |      |          | % |        |
|-------|-----|-----|----|-----------|------|----------|---|--------|
```

Figura 12.8

12.5 Analisi dei dati richiesti in OUTPUT

12.5.1

Si nota che Giornale ed Estratto conto hanno lo stesso contenuto formale con la sola differenza che i dati del giornale sono i dati di tutte le operazioni effettuate, mentre quelli di un conto, pur ordinati cronologicamente, riguardano solo le operazioni classificabili come appartenenti al conto stesso.

12.5.2

Si nota ancora che i dati in questione sono di due tipi (vedi fig. 12.9) :

```
 dati variabili operazione per operazione
-----------------------------------------------------------------------
|Data | Num.|Data | Num.|              |           |    Importi      |
| reg.|N.reg| doc.| doc.|              |           | DARE   | AVERE  |
|-----|-----|-----|-----|--------------|-----------|--------|--------|

dati variabili scelti da un insieme di dati costanti.
-----------------------------------------------------------------------
|     |     |     |     | Descrizione  | Conto     |                 |
|     |     |     |     |              |           |        |        |
|-----|-----|-----|-----|--------------|-----------|--------|--------|
```

Figura 12.9

- dati variabili operazione per operazione
- dati variabili operazione per operazione, ma scelti da un insieme di dati costanti

Si nota che la Descrizione è scelta da un insieme di descrizioni possibili elencate in una apposita tabella, che la denominazione del conto è scelta sulla base della appartenenza della operazione effettuata ad uno dei conti del Piano.

Per quanto attiene ai registri delle fatture si nota che essi non abbisognano di dati diversi da quelli del giornale se non per quanto riguarda le note e le aliquote IVA; si tratta in ambedue i casi di dati scelti tra quelli di una apposita tabella.

Per quanto riguarda i mastrini dei conti si osserva che una parte dei dati in essi contenuti è costante.

C/G	Tipo	Denominazione conto		
fig. 12.10				

Figura 12.10

Una parte cambia in seguito ad idonei aggiornamenti.

			Totale DARE	Totale AVERE

Figura 12.11

CAPITOLO 13

CONTABILITA' GENERALE: GLI INPUT

13.1 La definizione dei file di input: MOVIMENTI, CONTI, PROFILE.
13.2 Le relazioni tra INPUT ed OUTPUT.
13.3 il file MOVIMENTI:
13.3.1 Le relazioni tra i record del file MOVIMENTI;
13.3 2 Le relazioni tra il file MOVIMENTI e gli altri file;
13.3.3 Il tracciato record del file MOVIMENTI.
13.4 Il file conti.
13.5 Le relazioni tra file MOVIMENTI e file CONTI in un esempio di funzionamento.

13.1 La definizione dei file di INPUT: MOVIMENTI, CONTI, PROFILE

Da quanto sopra risulta che i dati necessari agli output di una procedura di contabilità possono essere registrati in un unico file di dati variabili, per il Giornale generale, il Registro delle fatture, le schede dei movimenti contabili,in file di dati costanti per i dati costanti dei conti e le tabelle di causali, aliquote e note IVA.

Notiamo che per quanto riguarda i conti in rapporto al livello di analisi potremmo avere un diverso contenuto di dati costanti. Così se per conti di mastro o di gruppo possono essere sufficienti i dati appena descritti, molto più ampio e dettagliato è l'insieme dei dati necessario per i conti analitici e i conti delle anagrafiche Clienti e Fornitori.

Chiamiamo MOVIMENTI il file di dati temporanei, CONTI il file dei dati costanti inerenti i conti. In una o più tabelle registrate in appositi file o in un Profile si possono registrare i dati delle causali dei movimenti delle note e delle aliquote IVA.

13.2 Le relazioni tra INPUT ed OUTPUT.

Si individuano le seguenti relazioni dirette tra OUTPUT e file di INPUT:

```
OUTPUT                                          INPUT

Giornale generale                              file MOVIMENTI
Registri IVA
Schede di movimenti di conto

Mastri di conto                                file CONTI

Tabelle di causali                             PROFILE
Tabelle IVA
```

Figura 13.1

13.3 Il file MOVIMENTI

13.3.1

La esigenza di raggruppare sotto uno stesso conto i movimenti che gli appartengono impone di collegare tra di loro i movimenti della stessa categoria. Ciò può essere fatto mediante due puntatori *last*, *next* di cui il primo consente di risalire al precedente movimento dello stesso conto, il secondo al successivo.

13.3.2

Tra file MOVIMENTI e gli altri file si individuano le seguenti relazioni:

```
-------------------------------------------------------------------
|Data | Num.|Data | Num.| Descrizione  | Conto     |    Importi      |
| reg.|N.reg| doc.| doc.|              |           | DARE   | AVERE  |
|-----|-----|-----|-----|--------------|-----------|--------|--------|
    collegati  con        collegato con      collegati con
i  movimenti della       tavola causali              file CONTI

  stessa categoria
-----------
| Note |IVA|
|      | % |
|------|---|
collegati con
 tavole IVA
```

Figura 13.2

I collegamenti tra file e file possono essere effettuati mediante puntatori. Le tabelle possono essere tenute su disco, e allora la lettura verrà effettuata da disco con puntatori, oppure possono essere tenute in memoria centrale, dopo una prima lettura, e in

questo caso l' accesso alle singole voci di tabella deve essere effettuato mediante lettura di variabili multiple in memoria centrale.

Se il piano dei conti considera diversi livelli di osservazione della gestione allora è necessario definire anche i vincoli tra le diverse categorie di conti di gruppo, di mastro e sottoconti.

13.3.3

Per quanto detto il tracciato record di un file MOVIMENTI può risultare come il seguente:

```
file MOVIMENTI
                tracciato record
-------------------------------------------------------------------
|    |N.|    |N.|p.|p.|c |                    |        |p.|p.|c |c |
|d   |  |d   |  |  |S |o |                    |        |  |  |o |o |
|a   |d |a   |r |c |o |d.|                    |variazio|l |n |d.|d.|
|t d |o |t r |e |o |o |t |D |  Descrizione    |ne      |a |e |  |  |
|a o |c.|a e |g.|n |t |e |                    |        |s |x |E |A |
|  c.|  |  g.|  |t |o |s |                    |        |t |t |s |l |
|    |  |    |  |o |C/|c.|                    |        |  |  |e.|i.|
|----|--|----|--|--|--|--|--------------------|--------|--|--|--|--|
|  4 | 2|  4 | 2| 2| 2| 2|                 20 |      8 | 2| 2| 2| 2|
|____|__|____|__|__|__|__|____________________|________|__|__|__|__|

per un totale di 54 caratteri.
```

Figura 13.3

E' prudente aggiungere qualche campo di riserva per flag e altro portando la lunghezza del record a 64 caratteri.

I nomi delle variabili utilizzate per la definizione dei campi saranno le seguenti:

```
-------------------------------------------------------------------
| D  |N | D  |N |P |P |C |                    |        |L |N |C |C |
| D  |D | R  |R |C |S |D |       DSC$         |  IM$   |S |X |I |A |
| O  |O | E  |E |O |O |E |                    |        |T |T |V |L |
| C  |C | G  |G |N |T |S |                    |        |$ |$ |$ |$ |
| $  |$ | $  |$ |$ |$ |$ |                    |        |  |  |  |  |
|----|--|----|--|--|--|--|--------------------|--------|--|--|--|--|
```

Figura 13.4

I nomi delle variabili il programma saranno i seguenti:

D	N	D	N	P	P	C			L	N	C	C
D	D	R	R	C	S	D	DESC$	IM#	S	X	J	A
O	O	E	E	%	C	S			T	T	V	L
C	%	G	%		%	%			%	%	%	%

Figura 13.5

Per quanto riguarda il file CONTI su evidenziano le seguenti relazioni:

C/				
G		Denominazione	Totale	Totale
r	T			
u	i	conto	DARE	AVERE
p	p			
p	o			
o				

con il conto di gruppo con i movimenti
e con i sottoconti.

Figura 13.6

Il collegamento con il conto di gruppo cui il conto appartiene può essere stabilito mediante un puntatore.
Il collegamento con i sottoconti può essere stabilito mediante la registrazione del numero dei sottoconti di un dato conto e un puntatore.

Il collegamento con il file MOVIMENTI può essere definito mediante l'impiego di due puntatori *next, last*, alla prima e alla ultima registrazione di conto che interessa.

13.4 Il file CONTI

```
file CONTI
            tracciato record
-------------------------------------------------------------
|p.|n.|p |                     |p.|p.|         |         |
|G |S.|o | Denominazione       |p |u |Totale   |Totale   |
|r |C |i |                     |r |l |         |         |
|u |o |n |        conto        |i |t |  DARE   | AVERE   |
|p |n |t |                     |m |i |         |         |
|p |t |e |                     |o |m |         |         |
|o |i |r |                     |  |o |         |         |
|--|--|--|---------------------|--|--|---------|---------|
|2 |2 | 2|                 20 |2 |2 |       8 |       8 |
|__|__|__|_____________________|__|__|_________|_________|

Per un totale di 46 caratteri.
```

Figura 13.7

E' prudente tenere liberi altri 10 caratteri di riserva per eventuali flag ed altro.

I nomi delle variabili di campo saranno i seguenti:

```
-------------------------------------------------------------
|P |N |P |                     |P |P |         |         |
|G |S |S |    DENCO$           |P |U |  DAR$   | AVE$    |
|R |C |C |                     |R |L |         |         |
|U |O |O |                     |I |T |         |         |
|$ |$ |$ |                     |$ |$ |         |         |
|--|--|--|---------------------|--|--|---------|---------|
```

Figura 13.8

I nomi delle variabili di programma saranno i seguenti:

```
-------------------------------------------------------------
|P |N |P |                     |P |P |         |         |
|G |S |S |    DEN$             |R |U |    D#   |    A#   |
|R |C |C |                     |I |L |         |         |
|% |% |% |                     |% |% |         |         |
|--|--|--|---------------------|--|--|---------|---------|
```

Figura 13.9

13.5 Le relazioni tra file CONTI e file MOVIMENTI in un esempio di funzionamemto.

Disponendo di linguaggi senza accesso per chiave si costruisce una struttura a liste con le seguenti caratteristiche:

- le registrazioni dei singoli movimenti vengono fatte cronologica mente in un file MOVIMENTI in record successivi;

- ogni registrazione di movimento di un dato conto viene legata alla successiva dello stesso conto da un puntatore;

- ogni conto è legato alle movimentazioni che lo riguardano da uno o più puntatori.

Si disponga di un piano con i seguenti conti e numeri di riferimento (vedi figura 13.10):

CASSA	1	FORNITORI	6	MERCE C/ ACQUISTI	9
CLIENTI	2	CAMBIALI PASSIVE	7	MERCE C/ VENDITE	10
BANCA C/C	3	IVA SU VENDITE	8	ETC..	
CAMBIALI ATTIVE	4				
IVA SU ACQUISTI	5				

prospetto 13.10

Figura 13.10

(Si vedrà in altra sede come i numeri di riferimento possano essere disposti più razionalemente.)

Si debbano registrare il giorno 4/3 le seguenti operazioni:

1) riscossione di un credito verso CLIENTI di lire 500.000 in contanti; data rilascio ricevuta 3/3.

2) pagamento di un debito verso FORNITORI di lire 100.000 in contanti; data ricevuta 3/3

3) emissione di una fattura di 600.000 su CLIENTI IVA 18; data documento 4/3.

In un contabilità tenuta manualmente si fanno per le due operazione le seguenti registrazioni in giornale:

```
n. C/D|n. C/A|n.reg|                                |Parziali|Generali|
      |      |    1 ------------4/3/84------------|        |        |
    1 |    2 |      CASSA              a CLIENTI  |        | 500.000|
      |      |      riscosso credito da cliente in|        |        |
      |      |      c/to fattura n.14  su Bianchi |        |        |
      |      |    2 ------------4/3/84------------|        |        |
    6 |    2 |      FORNITORI          a CASSA    |        | 300.000|
      |      |      pagato fornitore in conto fat=|        |        |
      |      |      tura n.9 Neri                 |        |        |
      |      |      ------------4/3/84------------|        |        |
    2 |      |    3 CLIENTI            a DIVERSI  |        | 708.000|
      |      |      ns. fattura n.17 su Rossi     |        |        |
      |     10|                         C.VENDITE A | 600.000|        |
      |      8|                         IVA VENDITE | 108.000|        |
      |      |      ------------       ------------|--------|        |
```

Figura 13.11

Dopo le registrazioni i mastri presentano il seguente contenuto:

```
  n.1 CONTO CASSA                      n.6 CONTO FORNITORI
-----------------------------        -----------------------------
n.r|          |          |n.r        n.r|          |          |n.r
   | x.xxx.xxx|          |              |          |xx.xxx.xxx|
  1|   500.000|  300.000| 2           2|  300.000|          |
   |          |          |              |          |          |

  n.2 CONTO CLIENTI                    n.8 CONTO IVA VENDITE
-----------------------------        -----------------------------
n.r|          |          |n.r        n.r|          |          |n.r
   |xx.xxx.xxx|  500.000| 1             |          |xx.xxx.xxx|
  3|   708.000|          |              |          |   108.000| 3
```

Figura 13.12

Il contenuto del file MOVIMENTI è dopo le prime due registrazioni il seguente:

data doc.	N. doc.	data reg.	N. reg.	p. conto	p. Sotto C/	cod. Desc.	Descrizione	variazione	p. last	p. next	cod. Ese.	cod. Ali.	
4/3	14	3/3	1	1	1	3	pagamento clienti	500.000	0	4			1
4/3	15	3/3	1	2	1	3	pagamento clienti	-500.000	0	5			2
4/3	9	4/3	2	6	2	4	pagamento fornitori	300.000	0	0			3
4/3	10	4/3	2	1	1	4	pagamento fornitori	-300.000	1	0			4
4/3	17	4/3	3	2	3	1	ns fattura	708.000	2	0			5
4/3	17	4/3	3	10	1	1	ns fattura	-600.000	0	0	1	3	6
4/3	17	4/3	3	8	1	1	ns fattura	-108.000	0	0	1	3	7

Figura 13.13

p. Gruppo	n. S. Conti	p. Ointer	Denominazione conto	p. primo	p. ultimo	Totale DARE	Totale AVERE
1	n	z	Cassa	1	4	500.000	300.000
2	2	x	Clienti	2	5	708.000	500.000
6		y	Fornitori	3	3	300.000	

Figura 13.14

CAPITOLO 14

IL PROGRAMMA DI APERTURA DEI CONTI

14.1 Caratteristiche del programma
14.2 Caratteristiche dell'Indice
14.3 Il programma:
14.3.1 le operazioni da eseguire;
14.3.2 il diagramma di struttura;
14.3.3 la pseudocodifica.
14.3.4 la codifica.

14.1 Caratteristiche del programma

Definita la struttura dei file di dati costanti CONTI, si deve progettare un programma di apertura e registrazione.

Si considera la struttura dei dati in input e si decide di preparare un programma con le seguenti caratteristiche:

- i conti di gruppo e i conti di mastro hanno lo stesso tracciato record;

- ogni conto di gruppo è legato mediante puntatori al primo dei conti di mastro che gli appartiene;

- ogni conto di gruppo ha un numero di conti che gli appartengono definito dall'operatore;

- l'operatore deve provvedere a predisporre dei conti in bianco per eventuali espansioni o modifiche del piano

- un indice gestisce i puntatori nel file.

14.2 Caratteristiche dell'Indice

L'indice deve consentire di risalire al contenuto di un dato conto mediante chiavi numeriche (del tipo ng.nm) con cui si identificano i conti di gruppo e i conti di mastro. Inoltre deve dare le indicazioni sul numero delle registrazioni effettuate e sul numero dei record occupati nel file MOVIMENTI alla fine di ogni registrazione per consentire la effettuazione della registrazione successiva.

Per tali scopi è sufficiente impiegare un file di record con un campo alfabetico, per le denominazioni dei campi di gruppo, dei campi numerici per la definizione del numero di conti appartenenti e di puntatori al primo di essi.

Il primo record dell'indice viene in particolare destinato ad accogliere i dati inerenti il nome del file di indice, il numero dei conti di gruppo e il numero delle registrazioni in giornale.

I record successivi costituiscono invece l'indice vero e proprio.

L'INDICE può essere costruito secondo lo schema della fig. 14.1:

Record	Nome File	N. Conti di Gruppo	N. Record occupati	N. Registrazioni	N. Movimenti
1	CONTI	5		0	0

	Titolo conti di GRUPPO	N. Conti di MASTRO	Pointer I conto	N. Sotto Conti	Pointer I S.conto
2	ATTIVITA'	30	6	1000	1
3	PASSIVITA'	30	36	400	1001
4	CAPITALE	5	66	30	1401
5	COSTI	20	71	60	1431
6	RICAVI	20	91	60	1491

Figura 14.1

Il tracciato record è il seguente:

Denominazione File al primo record	N. Conti	N. Record	N. Registrazioni	N. Movimenti
Titolo conti ai successivi				
20	2	2	2	2

per un totale di 28 caratteri.

Figura 14.2

I nomi delle variabili impiegate per la definizione dei campi sono:

```
-----------------------------------
|                    |N |P |L |L |
|       T$           |$ |$ |1 |2 |
|                    |  |  |$ |$ |
|--------------------|--|--|--|--|
```

Figura 14.3

I nomi delle corrispondenti variabili di programma sono:

```
-----------------------------------
|                    |N |P |L |L |
|       TJ$          |% |U |1 |2 |
|                    |  |% |% |% |
|--------------------|--|--|--|--|
```

Figura 14.4

14.3 Il programma:

14.3.1 Le operazioni da eseguire.

Le operazioni che il programma deve eseguire sono:

- apertura e dimensionamento file CONTI e INDICE e intestazione;
- input dati conto di gruppo;
- accumulo numero conti di gruppo;
- input dati conto di mastro;
- registrazione conto di mastro;
- determinazione puntatori;
- registrazione conto di gruppo;
- registrazione indice.

14.3.2 Il diagramma di struttura.

La struttura del programma è quindi la seguente:

Figura 14.5

14.3.3 La pseudocodifica.

La pseudocodifica del programma è quindi:

Blocco A) APERTURA PIANO DEI CONTI

APRI i file CONTI e INDICE
INPUT dati di intestazione

GOSUB INPUT dati conti di gruppo
GOSUB INPUT dati conti di mastro e REGISTRAZIONE
GOSUB REGISTRAZIONE Indice

CHIUDI i file CONTI e INDICE

RETURN

Blocco b1) INPUT DATI CONTI DI GRUPPO

RIPETI

INCREMENTA il numero dei conti
INPUT dati conti di gruppo

FINCHE' non hai finito

RETURN.

Blocco b2) INPUT DATI CONTI DI MASTRO

```
RIPETI

    RIPETI

        INCREMENTA n.conti di Mastro
        INPUT dati conti di Mastro
        REGISTRA i conti di Mastro

    FINCHE' non hai più conti di mastro di un dato gruppo
    REGISTRA i conti di GRUPPO

FINCHE' non hai finito i conti di gruppo

RETURN.
```

Blocco b3) REGISTRAZIONE INDICE

```
REGISTRA il primo record per i dati generali di indice
    RIPETI
        REGISTRA i record successivi per i dati dei conti
    FINCHE' non hai finito i conti di Gruppo.
RETURN
```

Volendo espandere la apertura alla serie dei sottoconti per tutti i successivi livelli del **piano** si deve utilizzare il modulo seguente:

```
RIPETI
    LEGGI i conti
    RIPETI

        INPUT dati sottoconti
        REGISTRA sottoconti

    FINCHE' non hai altri sottoconti
    REGISTRA numero conti e puntatori
FINCHE' hai esaurito tutti i conti
```

14.3.4 La codifica.

Blocco A) (listato 14.1)

```
LISTATO 14.1

100 REM BLOCCO A) : APERTURA DEL PIANO DEI CONTI
110 GOSUB 12000:GOSUB 12800'APERTURA INDICE E
    CONTI
120 REM IL NOME DEI FILE UTILIZZATI E' FISSO
130 GOSUB 500'BLOCCO B 1) INPUT DATI CONTI DI GRUPPO
140 GOSUB 700'BLOCCO B 2) INPUT DATI CONTI DI MASTRO
    E REGISTRAZIONE
150 GOSUB 900'BLOCCO B 3) REGISTRAZIONE INDICE
160 CLOSE
170 RETURN
```

Blocco b1)

Nel Blocco b1) si impiegano le seguenti variabili (fig. 14.6):

Tavola delle variabili blocco b1)

Nome variabile	tipo	contenuto	n.caratteri
NG%	intero	numero dei conti di Gruppo	2
TI$	stringa	titolo conto di G	20
TIN$(NG%)	stringa	titolo di ciascun conto di Gruppo	20

Figura 14.6

La codifica è:

```
LISTATO 14.2

500 REM BLOCCO B 1) : INPUT DATI CONTI DI GRUPPO

510 GOSUB 17300'MASCHERA INPUT
520 NG%=NG%+1'INCREMENTA NUMERO DEI CONTI RIPETI
530 C=3:LOCATE V%(C),H%(C),0:PRINT NG%
```

```
540 C=4:GOSUB 11200:TI$=IN$(5):TIN$(NG%)=TI$'
    INPUT DATI CONTI DI GRUPPO
550 C=5:GOSUB 11200:AN$=IN$(5)
560 IF AN$="Y" THEN 520 'FINCHE' NON HAI FINITO

570 RETURN
```

Blocco b2)

Si impiegano nella codifica le seguenti variabili:

Tavola delle variabili blocco b2)

Nome variabile	tipo	contenuto	n.caratteri
NF%	intero	numero logico del file CONTI	2
NG%	intero	numero conti di G	2
CG%	intero	chiave numerica di un conto di Gruppo	2
TIN$(CG%)	stringa	titolo conto CG%	20
N%(CG%)	intero	numero dei conti di mastro del conto CG%	2
NSC%	intero	numero di conti	2
P%(CG%)	intero	puntatore al primo conto di CG%	2
PSC%	intero	puntatore al primo conto	2
PGR%	intero	puntatore al conto di Gruppo CG%	2
PRI%	intero	puntatore alla prima registrazione	2
PUL%		puntatore all'ultima registrazione	2
D#	doppia p	DARE	2
A#	doppia p	AVERE	
ADR%	intero	indirizzo di record	2

Figura 14.7

Il listato è:

LISTATO 14.3

```
700 REM BLOCCO B 2): INPUT DATI CONTI DI MASTRO

710 NF%=2:GOSUB 12800:ADR%=NG%:P(0)=NG%+1

720 FOR CG%=1 TO NG%'                          RIPETI
730 GOSUB 17400'MASCHERA DI INPUT DATI DI MASTRO
740 C=3:LOCATE V%(C),H%(C)+3,0:PRINT CG%
```

```
750 C=4:LOCATE V%(C),H%(C),0:PRINT TIN$(CG%)
760 N%(CG%)=N%(CG%)+1
770 C=5:LOCATE V%(C),H%(C),0:PRINT N%(CG%)
780 C=6:GOSUB 11200:DEN$=IN$(5)
790 NSC%=0:PGR%+CG%:PRI%=0:PUL%=0:D#=0:A#=0
800 ADR%=ADR%+1:R%=ADR%:GOSUB 12900:GOSUB 10600
810 C=7:GOSUB 11200:AN$=IN$(5)
820 IF AN$="Y" THEN 730'
            FINCHE' NON HAI PIU'CONTI DI MASTRO
830 REM REGISTRA CONTI DI GRUPPO
840 R%=CG%:DEB$=TIN$(CG%):NSC%=N%(CG%):P%(CG%)
    =P%(CG%-1)+N%(CG%-1):PSC%=P%(CG%)
850 GOSUB 12900:GOSUB 10600'CARICA BUFFER E
    REGISTRA
860 NEXT CG%'       PER TUTTI I CONTI DI GRUPPO

870 CLOSE #2

880 RETURN
```

Blocco b3) REGISTRAZIONE INDICE

Le variabili impiegate nel blocco b3) sono:

Tavola delle variabili blocco b3)

Nome variabile	tipo	contenuto	n.caratteri
NF$	stringa	nome file	20
N%	intero	al primo record numero conti di Gruppo	2
NG%	intero	numero conti di G.	2
ADR%	intero	numero record registrati in CONTI	2
CG%	intero	chiave conto di G.	2
N%(CG%)	intero	numero dei conti di di CG%	2
P%(CG%)	intero	puntatore al primo conto di CG%	2
PU%	intero	puntatore al primo conto	2
L1%	intero	numero registrazioni	2
L2%	intero	numero movimenti	2

Figura 14.8

Il listato è il seguente:

LISTATO 14.4

```
900 REM BLOCCO B 3): REGISTRAZIONE INDICE

910 TI$=NF$:N%=NG%:PU%=ADR%:L1%=0:L2%=0
920 R%=1:NF%=1:GOSUB 12000:GOSUB 12100:
    GOSUB 10600'  REGISTRA PRIMO RECORD

930 FOR CG%=2 TO NG%+1'              RIPETI
940 TI$=TIN$(CG%-1):N%=N%(CG%-1):PU%=P%(CG%-1)
    :L1%=0:L2%=0
950 GOSUB 12100:GOSUB 10600'
                        CARICA BUFFER E REGISTRA
960 NEXT CG%'      PER TUTTI I CONTI DI GRUPPO

970 CLOSE #1

980 RETURN
```

Si hanno qui di seguito i listati di apertura, carico buffer e assegnazione dei file INDICE, CONTI e MOVIMENTI (14.5, 14.6, 14.7).

```
12000 REM APERTURA INDICE
12010 NF$="INDICE"
12020 OPEN "R",#1, NF$,28
12030 FIELD #1,20 AS T$,2 AS N$,2 AS P$,2 AS L1$,2 AS L2$
12040 RETURN
```

```
12100 REM CARICO BUFFER FILE INDICE, REGISTRAZIONE INDICE
12110 LSET T$=TI$:RSET N$=MKI$(N%):RSET P$=MKI$(PU%)
12120 LSET L1$=MKI$(L1%):LSET L2$=MKI$(L2%)
12130 RETURN
```

```
12200 REM ASSEGNAZIONE INDICE
12210 TI$=T$:N%=CVI(N$):PU%=CVI(P$):L1%=CVI(L1$):L2%=CVI
      (L2$)
12220 RETURN
```

```
1
12800 REM FILE CONTI
12810 OPEN "R",#2,"CNTI",56
12820 FIELD #2,2 AS PGRU$,2 AS NSCO$,2 AS PSCO$,20 AS DENCO$,
       2 AS PRI$,2 AS PULT$,8 AS DAR$,8 AS AVE$,10 AS RIS$
12830 RETURN



12900 REM CARICO BUFFER FILE CONTI
12910 LSET PGRU$=MKI$(PGR%):LSET NSCO$=MKI$(NSC%):LSET PSCO$
      =MKI$(PSC%):LSET DENCO$=DEN$:LSET PRI$=MKI$(PRI%):LSET
      PULT$=MKI$(PUL%):LSET DAR$=MKD$(D#):LSET AVE$=MKD$(A#):
      LSET RIS$
12920 RETURN



12930 REM CARICO BUFFER CONTI IN REGISTRAZIONE CONTABILE
12940 LSET PRI$=MKI$(PRI%):LSET PULT$=MKI$(PUL%):LSET DAR$=
      MKD$(D#):LSET AVE$=MKD$(A#)
12950 RETURN



13000 REM ASSEGNAZIONE VARIABILI DI CAMPO,FILE CONTI.
13010 PGR%=CVI(PGRU$):NSC%=CVI(NSCO$):PSC%=CVI(PSCO$):DEN$=
      DENCO$:PRI%=CVI(PRI$):PUL%=CVI(PULT$):D#=CVD(DAR$):
      A#=CVD(AVE$)
13020 RETURN



13500 REM APERTURA FILE MOVIMENTI
13510 OPEN "R",#3,"MV",64
13520 FIELD #3,4 AS DDOC$,2 AS NDOC$,4 AS DREG$,2 AS NREG$,2
      AS PCON$,2 AS PSOT$,2 AS CDES$,20 AS DSC$,8 AS IM$,2 AS
      LST$,2 AS NXT$,2 AS CIV$,2 AS CAL$
13530 RETURN
```

```
13550 REM CARICO BUFFER MOVIMENTI
13560 LSET DDOC$=MKS$(DDOC):LSET NDOC$=MKI$(NDO%):LSET DREG$=
      MKS$(DREG):LSET NREG$=MKI$(NRE%):LSET PCON$=MKI$(PC%):
      LSET PSOT$=MKI$(PSC%):LSET CDES$=MKI$(CDS%)
13570 LSET DSC$=DESC$:LSET IM$=MKD$(IM$):LSET LST$=MKI$(LST%):
LSET NXT$=MKI$(NXT%):LSET CIV$=MKI$(CIV%):LSET CAL$=MKI$(CAL%)
13590 RETURN
```

```
13600 REM ASSEGNAZIONE FILE MOVIMENTI
13610 DDOC=CVS(DDOC$):NDO%=CVI(NDOC$):DREG=CVS(DREG$):NRE%=
     CVI(NREG$):PC%=CVI(PCON$):PSC%=CVI(PSOT$):CDS%=CVI(CDES$)
13620 DESC$=DSC$:IM$=CVD(IM$):LST%=CVI(LST$):NXT%=CVI(NXT$):
      CIV%=CVI(CIV$):CAL%=CVI(CAL$)
13630 RETURN
```

CAPITOLO 15

LA REGISTRAZIONE DEI MOVIMENTI CONTABILI

15.1 Oggetti di rilevazione in PRIMA NOTA;
15.2 Uno schema di PRIMA NOTA;
15.3 La registrazione dei movimenti contabili:
15.3.1 la pseudocodifica
15.3.2 le tavole delle variabili e i listati di programma

15.1 Oggetti di rilevazione in PRIMA NOTA

Per ottenere la certezza che nessun fatto amministrativo venisse trascurato, tradizionalmente si procedeva all'annotazione del fatto stesso nel momento in cui si verificava in un *brogliaccio* o in una *prima nota*. Detta registrazione risulta utile particolarmente per i movimenti di cassa e per tutti gli altri elementi dell'attivo e del passivo, al fine di controllo della corrispondenza tra consistenza effettiva e contabile.

Disponendo di procedure automatizzate la prima nota può non essere tenuta manualmente anche se viene di fatto sempre costruita e utilizzata.

Si considerino le fatture ricevute da Fornitori per evidenziarne gli elementi da registrare in partita doppia.(fig. 15.1)

```
                          FATTURA
-------------------------------------------------------------------
|                                                                 |
| DITTA O RAGIONE SOCIALE            Codice Fornitore         |  T  |
|  DEL FORNITORE                                              |  E  |
|                                 Registrazione:Data          |  S  |
|                                               numero        |  T  |
|                                                             |  A  |
| DATA FATTURA      NUMERO FATTURA     Numero attribuito            |
|-----------------------------------------------------------------|

|                                                             |  C  |
|                                                             |  O  |
|                                                             |  R  |
|                                                             |  P  |
|                                                             |  O  |
|-----------------------------------------------------------------|
|                                                             |  F  |
| Codice costo                                                |  I  |
|                                                             |  N  |
| IMPONIBILE     ALIQUOTA IVA    IMPORTO IVA   TOTALE FATTURA |  E  |
|-----------------------------------------------------------------|
```

Figura 15.1

Gli elementi delle fatture da registrare in partita doppia sono:

a) RAGIONE SOCIALE O DITTA FORNITRICE;
b) numero e data di registrazione;
c) NUMERO E DATA DEL DOCUMENTO;
d) conto;
e) IMPORTO;
f) IMPORTO IVA;
g) TOTALE FATTURA;

Per le registrazioni IVA sono necessari i seguenti elementi:

h) IMPONIBILE;
i) ALIQUOTA;
l) assoggettamento;
m) numero documento.

15.2 Schema di PRIMA NOTA

In una procedura automatizzata una parte degli elementi da registrare viene immessa per lettura dei dati dei documenti di prova, una parte viene decisa dall'operatore mediante selezione di codici opportuni. Spesso si attribuiscono i codici direttamente sul documento e si passano i documenti codificati alla registrazione. Gli elementi che nel modello di fattura sono scritti in maiuscolo sono originariamente contenuti nel documento stesso, quelli in lettere minuscole debbono essere attribuiti da qualche impiegato prima della registrazione.

Normalmente la rilevazione si sviluppa su tre conti, quello inerente il conto di costo da addebitare, quello inerente l'IVA, quello inerente il Fornitore.

Si può pertanto utilizzare il seguente schema di registrazione su tre righe di prima nota:

PRIMA NOTA

data doc.	N. doc.	data reg.	N. reg.	cod. operaz.	cod. conto DARE	cod. conto AVERE	Descrizione eventuale	Importo Operazione	cod. Ese.	cod. Ali.
*	*		*	*	*costo			*		
					*c.IVA			*	*	*
						*C.For		*		

Figura 15.2

Si fa osservare che l'operatore non dovrebbe procedere alla registrazione di tutti gli elementi delle tre righe, ma alla registrazione delle sole parti segnate da asterisco in quanto le altre, essendo ripetizione di dati della riga precedente, possono essere duplicate automaticamente.

15.3 La registrazione dei movimenti contabili.

La registrazione dei movimenti contabili richiede in primo luogo la apertura dei file MOVIMENTI e CONTI e la lettura in MEMORIA dell'**indice**. La lettura in memoria dell'**indice** consente di avere in variabili di memoria i puntatori ai record con un risparmio di tempo di ricerca rispetto ad altre soluzioni.

Trattandosi di registrazioni in partita doppia notiamo che ognuna di esse è composta di una o più registrazioni in DARE e di una o più registrazioni in AVERE bilancianti in modo che il totale del DARE equivalga al totale dell'AVERE.

Per ogni registrazione devono essere visualizzate le richieste di *input* del programma che corrispondono ai dati richiesti dalla *primanota*. Alcuni di questi elementi però non debbono essere ripetuti; il numero della registrazione deve essere calcolato da programma, le date di registrazione successive alla prima della giornata debbono essere ripetute automaticamente.

15.3.1

La pseudocodifica di un programma di registrazione potrebbe essere fatta nel modo seguente:

```
Blocco A): INIZIALIZZAZIONE

DIMENSIONA
APRI il file CONTI
APRI il file MOVIMENTI
APRI il file INDICE
LEGGI il file INDICE
CHIAMA il blocco di CONTROLLO delle REGISTRAZIONI
FINE.


Blocco B): CONTROLLO delle REGISTRAZIONI

    RIPETI
    SEGNALA che sei in DARE
        1 RIPETI

            CHIAMA l'INPUT di PRIMANOTA
            DEFINISCI il SEGNO della variazione di INPUT
            CHIAMA la REGISTRAZIONE
            CONTROLLA LO SBILANCIO DARE AVERE
```

FINCHE' hai movimenti DARE
SEGNALA che sei in AVERE
RIPRENDI dal punto 1
FINCHE' non hai più registrazioni
REGISTRA numero di registrazioni sull'INDICE
RETURN

Blocco C1): INPUT DI PRIMANOTA

VISUALIZZA il numero di registrazione
DA' la data se vuoi
DA' tutti gli altri INPUT controllando il codice del conto
LEGGI il record del file CONTI che corrisponde al codice
SE il FLAG è DARE aumenta il DARE, se no aumenta l'AVERE di conto
RETURN

Blocco C2): REGISTRAZIONE MOVIMENTI E CONTI

INCREMENTA il numero di record registrati
ASSEGNA al puntatore di " registrazione precedente " il valore della " ultima registrazione di conto "
REGISTRA MOVIMENTI
ASSEGNA all"' ultima registrazione " il valore di "numero di record registrati " all'indirizzo di " codice conto " REGISTRA CONTI

RETURN

15.3.2
Le variabili di programma del blocco B sono le seguenti (fig. 15.3):

Tavola delle variabili blocco B)

Nome variabile	tipo	contenuto	n.caratteri
S%	intero	1 / -1	2
FL%	intero	1 o piu' di 1	2
DARE#	doppia p	dare di registrazio ne in video	8
AVERE#	doppia p	avere di registrazio ne in video	8
VC#	doppia p	variazione di conto	8

Figura 15.3

Il listato di programma è quindi il seguente (listato 15.1):

```
LISTATO 15.1

2000 REM BLOCCO A) INIZIALIZZAZIONE

2010 GOSUB 12000:GOSUB 12800:GOSUB 13500'APRI
                              INDICE-CONTI-MOVIMENTI

2020 GOSUB 1000'LEGGI INDICE

2030 GOSUB 2050'BLOCCO B) CONTROLLO REGISTRAZIONI

2040 RETURN


2050 REM BLOCCO B)CONTROLLO DELLE REGISTRAZIONI
                                          IN P.D.

2060 DARE#=0:AVERE#=0:FL%=0:GOSUB 17500'
                         MASCHERA DI INPUT   (1)

2070 S=1'                SEGNALA CHE SEI IN DARE

2080 GOSUB 2200'   INPUT DI PRIMANOTA       (2)

2090 VC#=VC#*S:IM#=VC#:BIL#=BIL#+VC#'CONTROLLA
                                 LO SBILANCIO
2100 GOSUB 2500'                 REGISTRAZIONE
2110 C=14:GOSUB 11200:AN$=IN$(5)' CHIEDI SE HAI
                                      MOVIMENTI
2120 IF AN$="Y" THEN 2080'FINCHE' HAI MOVIMENTI
                               TORNA AL PUNTO 2


2130 TMS$(10)="AVERE":S=-1:C=10:
     LOCATE V%(C),1,0:PRINT"      "
2135 REM                SEGNALA CHE SEI IN AVERE


2150 IF AN$="N" AND FL%=1 THEN  2080'  RIPRENDI
                                    DAL PUNTO 2

2160 NRE%=NRE%+1'    INCREMENTA IL NUMERO DELLE
                     REGISTRAZIONI PER L'INDICE
```

```
2170 C=17:GOSUB 11200:AN$=IN$(5):IF AN$="Y"
     THEN 2060'FINCHE' NON HAI FINITO RIPPENDI
                                        DAL PUNTO 1

2180 L1%=NRE%:L2%=REC%:NF%=1:R%=1:GOSUB 10500:
     GOSUB 12120:GOSUB 10600
2185 GOSUB 2600'       RIORDINA I PUNTATORI NXT%

2190 RETURN
```

Le variabili di programma da impiegare nel blocco C1), oltre quelle definite in sede di lettura del file MOVIMENTI e CONTI e quelle tipiche di gestione maschere, sono le seguenti:

Tavola delle variabili blocco C1)

Nome variabile	tipo	contenuto	n.caratteri
CG%	intero	chiave numerica di un conto di gruppo	2
CO%	intero	chiave numerica di un conto di mastro	2
CS%	intero	chiave numerica di un sottoconto	2
NG%	intero	numero dei conti di gruppo	2
N%(CG%)	intero	numero dei conti di mastro di un conto di gruppo	2
P%(CG%)	intero	puntatore al primo dei conti di mastro del conto CG%	2

fig. 15.4

Figura 15.4

Il listato di programma è il seguente:

LISTATO 15.2

```
2200 REM BLOCCO C1): INPUT DI PRIMANOTA

2205 RESTORE
2210 C=1:LOCATE V%(C),H%(C),0:PRINT NRE%+1
2220 C=2:LOCATE V%(C),H%(C),0:GOSUB 10100:IF
IN$(5)<>CHR$(13) THEN GOSUB 10200:DREG=GIORD
2230 C=3:GOSUB 10100:GOSUB 10200:DDOC=GIORD'

PRENDE DA TASTIERA I DATI DEI MOVIMENTI

2240 C=4:GOSUB 11200:NDO%=VAL(IN$(5))
2250 C=5:GOSUB 11200:CDS%=VAL(IN$(5))
2260 C=6:GOSUB 11200:CG%=VAL(IN$(5))
2270 IF CG%<1 OR CG%>NG% THEN 2260 '

PRENDE DA TASTIERA I DATI DI CONTO DI GRUPPO E
DI MASTRO E LI CONTROLLA

2280 C=7:GOSUB 11200:CO%=VAL(IN$(5))
2290 IF CO%<1 OR CO%>N%(CG%) THEN 2280
2300 R%=P%(CG%)+CO%-1:NF%=2:GOSUB 10500:
     GOSUB 13000' LEGGI IL RECORD DI CONTI


2310 C=8:LOCATE V%(C),H%(C),0:PRINT DEN$'


     VISUALIZZA E POI CHIEDE LA DESCRI=
     ZIONE.

2320 C=9:GOSUB 11200:DESC$=IN$(5)
2330 C=11:GOSUB 11200:VC#=VAL(IN$(5))
2340 PC%=R%:'

     Con la prossima riga se il flag e' DARE
       aumenta il DARE,se no aumenta l'AVERE

2350 IF TMS$(10)="DARE" THEN D#=D#+VC#:DARE#=
   DARE#+VC# ELSE A#=A#+VC#:AVERE#=AVERE#+VC#
2360 C=15:LOCATE V%(C),H%(C),0:PRINT DARE#
2370 C=16:LOCATE V%(C),H%(C),0:PRINT AVERE#
2380 C=12:GOSUB 11200:CIV%=VAL(IN$(5))
2390 C=13:GOSUB 11200:CAL%=VAL(IN$(5))

2400 RETURN
```

Il listato di programma del blocco C2) è il seguente (listato 15.3):

```
LISTATO 15.3

2500 REM BLOCCO C2): REGISTRAZIONE MOVIMENTI E
                                          CONTI
2505 LOCATE 23,50,0

2510 REM REGISTRAZIONE MOVIMENTI

2520 NF%=3:REC%=REC%+1:R%=REC%:LST%=PUL%
2530 GOSUB 13550:GOSUB 10600

2540 REM REGISTRAZIONE AGGIORNAMENTO CONTI
2550 NF%=2:R%=P%(CG%)+CO%-1:PUL%=REC%:IF PRI%
     =0 THEN PRI%=PUL%
2560 GOSUB 12930:GOSUB 10600

2570 RETURN

2600 REM RIORDINA PUNTATORI NXT%

2605  GOSUB 1000:GOSUB 12800:GOSUB 13500'
          LEGGI INDICE E APRI MOVIMENTI E CONTI

2610 FOR CG%=1 TO NG%'                    RIPETI

2620 FOR CO%=1 TO N%(CG%)'                RIPETI

2630 R%=P%(CG%)+CO%-1'
                    ALL'INDIRIZZO COSI' CALCOLATO
2640 NF%=2:GOSUB 10500:GOSUB 13000'
                              LEGGI IL FILE CONTI

2650 WHILE PUL%>0'            MENTRE IL PUNTATORE
           ALL'ULTIMO MOVIMENTO E' POSITIVO (1)

2660 NF%=3:R%=PUL%:GOSUB 10500:GOSUB 13600'
             LEGGI L'ULTIMO MOVIMENTO DI CONTO
2670 IF LST%>0 THEN PRI1%=LST% '
   ASSEGNA A PRIMO IL VALORE DI PRECEDENTE(LST)
2680 R%=LST%:GOSUB 10500:GOSUB 13600'
                              LEGGI IL PRECEDENTE
2700 NXT%=PUL%:GOSUB 13550:GOSUB 10600:PUL%=LST%'


2710 WEND'                 RIPRENDI DAL PUNTO   (1)
```

```
2720 NF%=2:R%=P%(CG%)+CO%-1:GOSUB 10500:
     GOSUB 13000'                    RILEGGI CONTI
2730 PRI%=PRI1%:GOSUB 12900:GOSUB 10600'REGISTRA
                                IL PUNTATORE PRIMO

2740 NEXT CO%'        PER TUTTI I CONTI DI MASTRO

2750 NEXT CG%'        PER TUTTI I CONTI DI GRUPPO

2760 RETURN
```

CAPITOLO 16

LE STAMPE DEI DOCUMENTI DI PROVA E DEI REGISTRI DELLE OPERAZIONI DI ESERCIZIO

16.1 Schema generale dei diagrammi di struttura dell'output.
16.2 Le subroutine di stampa.
16.3 I programmi principali di visualizzazione o stampa:
16.3.1 del GIORNALE;
16.3.2 del BILANCIO DI VERIFICA;
16.3.3 dei MOVIMENTI DI CONTO;
16.3.4 dell'INDICE.

16.1 Schema generale dei diagrammi di struttura dell'output

Le stampe dei documenti di registrazione delle operazioni di esercizio hanno caratteristiche comuni: tutti i REGISTRI CONTABILI e molti *documenti originari* sono composti di una *intestazione*, di un *corpo*, di una *fine*. Fanno eccezione le TRATTE, gli ASSEGNI, le RICEVUTE e tutti quei documenti che non richiedono la ripetizione di righe simili per numero e tipo di campi.

La *intestazione* è costituita dai fondamentali dati anagrafici della ditta che presenta il documento e dal titolo del documento stesso.

Il *corpo* è, per molti dei documenti da stampare, composto di un numero variabile di pagine, composte a loro volta di un numero variabile di righe che può raggiungere un massimo definito. Ciascuna pagina è composta di una riga di testa costante e può riportare alla fine una riga di totali.

Dette righe o riportano in ordine cronologico tutte le operazioni compiute e registrate, o una parte di queste opportunamente scelte.

La *fine* è costituita di solito da una riga di totali opportunamente calcolati. Pertanto il diagramma di struttura risulta essere il seguente:

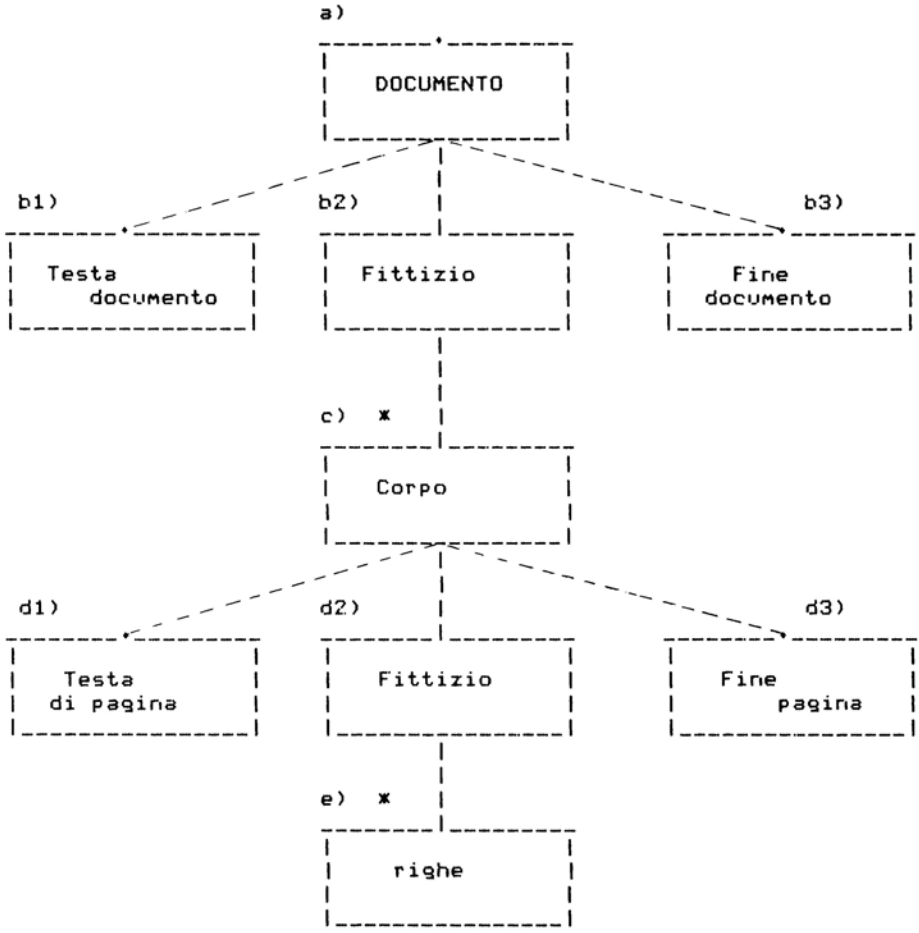


Figura 16.1

16.2 Le subroutine di stampa

Si consideri in particolare la parte riguardante la stampa di una pagina. Essa ,si è visto, è composta di una *testata*, di un *corpo,* di una *fine*.

La *testata* è composta di almeno due righe di cui una serve al titolo e al numero della pagina, l'altra alla descrizione dei campi delle righe i cui dati verranno stampati nelle righe successive.

Sono dati costanti per ciascuna pagina i dati seguenti:

- il titolo di pagina;
- la descrizione dei campi di riga;
- il numero delle righe per ciascuna pagina;

Il *corpo* è costituito di tante righe con un uguale numero di campi. Dovranno essere definiti i seguenti dati costanti per ciascuna riga:

- il numero dei campi per riga di stampa;
- il tipo delle variabili di campo di ciascuna riga;
- la *immagine di stampa* di riga.

La *fine* è costituita di almeno una riga con campi per i totali di pagina se vi sono righe con campi numerici sommabili.
I dati dell'ultima riga di ciascuna pagina possono richiedere i seguenti elementi costanti:

- una *immagine di stampa* per i totali di pagina.

Detti dati costanti possono essere definiti in apposite subroutine eventualmente distinte per *testata, corpo, fine*.

Le variabili da impiegare per dette subroutine possono avere i seguenti nomi:

Tavola delle variabili per Testa pagina

Nome variabile	tipo	contenuto	n.caratteri
TIT$	stringa	titolo di ogni pag.	x
LIN$	stringa	riga di separazione	x
TESPAG$	stringa	descrizione di ogni campo di riga	y
NCAMP%	intero	numero campi per riga;	
NPAG%	intero	contatore pagine	
NR%	intero	contatore numero riga.	
NRT%	intero	numero riga di testa	

Figura 16.2

Per la stampa si impiegheranno le istruzioni di LPRINT e LPRINT USING che permettono di stampare dati liberamente o su formato predefinito.

Per le righe del corpo di ogni pagina sono necessarie, oltre alle variabili già viste, variabili cui assegnare le immagini di stampa di ciascun campo di riga.

Tavole variabili di riga

Nome variabile	tipo	contenuto	n.caratteri
RIG$(I)	stringa	immagine di stampa del campo (I)	z
VRIG$(I)	stringa	dato da stampare con l'immagine (I)	z
FRIG$(I)	stringa	immagine di stampa del campo (I) nella ultima riga	k
VFRIG$(I)	stringa	dato da stampare nel campo (I)	k

Figura 16.3

Il programma per la stampa del corpo Blocco C) deve provvedere:

- a incrementare il numero di righe NR per ogni riga stampata
- a stampare la *testa* mediante le già viste istruzioni LPRINT qualora il numero di riga sia uguale a 1;
- a formare le righe con i dati voluti tenendo conto dei campi numerici e alfabetici necessari;
- a stampare la *fine* se è stato raggiunto il massimo delle linee consentite per pagina.

La subroutine che provvede alla stampa della *testa* della pagina ad ogni inizio pagina, delle righe successive e della *fine* ad ogni fine pagina ha la seguente codifica:

LISTATO 16.1

```
11800 REM STAMPA TESTA PAGINA E RIGHE
11810 REM VOGLIO IL NUMERO DEI CAMPI (NCAMP),
                 IL NUMERO DELLE RIGHE (NR)
11820 REM IL NUMERO DELLE RIGHE DI INTESTAZIONE
                      )NRT,IL NUMERO DELLA PAG.

11830 NR%=NR%+1'     INCREMENTA  POI STAMPA TESTA
11840 IF NR%=1 THEN PRINT TIT$:PRINT LIN$:PRINT
      TESPAG$:PRINT LIN$
```

```
11850 IF NR%=NMR% THEN 11860 ELSE GOTO 11900'
      SE LE RIGHE SONO FINITE STAMPA FINE PAGINA
      ALTRIMENTI STAMPA RIGA

11860 FOR I=1 TO NCAMP%'RIPETI PER TUTTI I CAMPI
      DI ULTIMA RIGA
11870 IF VFRIG$(I)="" THEN PRINT USING FRIG$(I);
      VFRIG$(I);ELSE IF VFRIG$(I)<>"" THEN PRINT
      USING FRIG$(I);VFRIG$(I);'
11880 NEXT I'Va al prossimo campo di ultima riga

11890 PRINT"INSERISCI UN NUOVO FOGLIO E PREMI UN
      TASTO":AN$=INPUT$(1):NR%=0:NPAG%=NPAG%+1:
      PRINT "Pag.n"NPAG%;TIT$:GOTO 11830'

11900 FOR I= 1 TO NCAMP%'
      PER TUTTI I CAMPI DI RIGA DI CORPO RIPETI
11910 IF VRIG$(I)="" THEN PRINT USING RIG$(I);
      VRIG$(I);ELSE IF VRIG$(I)<>"" THEN PRINT
      USING RIG$(I);VRIG$(I);'
11920 NEXT I'     Va al prossimo campo di riga

11930 PRINT'          SALTA ALLA RIGA SUCCESSIVA

11940 RETURN
```

La subroutine che provvede alla stampa del *corpo* può essere utilizzata anche per la stampa della *testa* del documento. Si utilizza invece una subroutine diversa per la stampa della *fine*.

La codifica di quest'ultima è:

LISTATO 16.2

```
11950 REM                          STAMPA FINE STAMPA

11960 FOR I=1 TO NFCAMP%'
                        PER I CAMPI DI FINE RIPETI

11970 IF VFDOC$(I)="" THEN PRINT USING FDOC$(I);
      VFDOC$(I);ELSE IF VFDOC$(I)<>"" THEN PRINT
      USING FDOC$(I);VFDOC$(I);'

11980 NEXT I'VA AL PROSSIMO CAMPO DI RIGA FINALE

11990 RETURN
```

16.3 I programmi principali di visualizzazione o stampa

Si presentano qui di seguito alcuni *programmi principali* che non richiedono stampa di dati di *testa documento* essendo sufficienti quelli di *testa riga*.

16.3.1

Il primo programma relativo al GIORNALE, provvede alla lettura di tutti i record del file MOVIMENTI scegliendo per la stampa i dati di movimento necessari

La sua codifica è la seguente:

```
LISTATO 16.3

3000 REM GIORNALE

3010 GOSUB 12800:GOSUB 13500:GOSUB 1000'
                                    LEGGI INDICE
3020 CLS: GOSUB 11700:GOSUB 18500'    DATI TESTA
3030 CR%=1:NF%=3'            PREPARA IL CONTATORE

3040 WHILE CR%<=REC%'      MENTRE IL CONTATORE E'

                    INFERIORE AL NUMERO DEI RECORD
3050 R%=CR%:NF%=3:GOSUB 10500:GOSUB 13600'
                      LEGGI IL RECORD DEL FILE MOV
3060 NF%=2:R%=PC%:GOSUB 10500:GOSUB 13000'
                                LEGGI IL FILE CONTI
3070  GIORD=DDOC:GOSUB 10300:DDOC=DDTA
3080  GIORD= DREG:GOSUB 10300:DREG=DDTA
3090 VRIG#(1)=NRE%:VRIG#(2)=DREG:VREG#(3)=NDO%:
     VRIG#(4)=DDOC'                       ASSEGNA
3100 VRIG$(5)= DEN$:VRIG$(6)=DESC$

3110 IF IM#>0 THEN VRIG#(7)=IM#:VRIG#(8)=0 ELSE
     VRIG#(7)=0:VRIG#(8)=IM#
3115 GOSUB 11800'                          STAMPA
3120 TOTD#=TOTD#+VRIG#(7):TOTA#=TOTA#+VRIG#(8)'
                                  CALCOLA TOTALI
3130 CR%=CR%+1

3140 WEND'                      RICOMINCIA DA WHILE

3150 VFDOC#(1)=TOTD#:VFDOC#(2)=TOTA#'     ASSEGNA
3160 GOSUB 11950'                 STAMPA FINE RIGA

3170 RETURN
```

16.3.2

Il secondo programma riguarda la stampa del BILANCIO DI VERIFICA eseguito sulla base della lettura dei dati registrati nel file CONTI.

La sua codifica è la seguente:

```
LISTATO 16.4

3500 REM   BILANCIO DI VERIFICA

3510 GOSUB 1000:CLS'                    LEGGI INDICE
3520 CLS:GOSUB 11700:GOSUB 18700'      DATI TESTA
3530 NF%=2'  DEFINISCI IL NUMERO LOGICO DEL FILE
                                              CONTI

3540 FOR CG%=1 TO NG%'      PER I GRUPPI   RIPETI

3550 FOR CO%=1 TO N%(CG%)' PER I MASTRI   RIPETI

3560 R%=P%(CG%)+CO%-1'     PRENDI L'INDIRIZZO DEL
                                              CONTO
3570 GOSUB 10500:GOSUB 13000
3580 VRIG$(1)=TI$:VRIG#(2)=D#:VRIG#(3)=A# '
                              ASSEGNA PER LA STAMPA
3590 TOTD#=TOTD#+D#:TOTA#=TOTA#+A#'  SOMMA PER I
                                             TOTALI

3600 NEXT CO%'        AL PROSSIMO CONTO DI MASTRO

3610 NEXT CG%'        AL PROSSIMO CONTO DI GRUPPO

3620 VFDOC$(1)="TOTALE:VFDOC#(2)=TOTD#:VFDOC#(3)
     =TOTA#'ASSEGNA
3630 GOSUB 11950'                       STAMPA FINE

3640 RETURN
```

16.3.3

Si presentano quindi programmi che richiedono la stampa di specifici dati di testa documento

Il primo, per la stampa, o la visualizzazione del movimento di un conto deve eseguire le seguenti operazioni fondamentali:

- presentare un menù di richiesta di conti da stampare;
- prendere i relativi dati di richiesta in input;
- leggere il conto richiesto e stamparne i dati in testa documento;
- passare alla lettura del file movimento dal record che corrisponde al puntatore di conto PRI% ai successivi puntati da NXT%;
- richiedere la stampa dei movimenti di conto;
- richiedere la stampa di *fine documento*.

La codifica è la seguente:

LISTATO 16.5

```
3700 REM   STAMPA ESTRATTO CONTO DEI CONTI DA  A

3710 GOSUB 13500:GOSUB 12800:GOSUB 1000:' APRI I
                              CONTI E LEGGI L'INDICE
3720 GOSUB 17600'                MASCHERA DI MENU'

3730 C=4:GOSUB 11200:CG%=VAL(IN$(5))'      CHIEDI
                                 IL CONTO DI  GRUPPO
3740 IF CG%<1 OR CG%>NG% THEN 3730 ' CONTROLLALO
3750 NF%=2:R%=CG%:GOSUB 10500:GOSUB 13000:CG1%=
     CG%'                        LEGGILO E ASSEGNA
3760 C=5:LOCATE V%(C),H%(C),0:PRINT DEN$

3770 C=6:GOSUB 11200:CO%=VAL(IN$(5)):CO1%=CO%'
                         CHIEDI IL CONTO DI MASTRO
3780 IF CO%<0 OR CO%>N%(CG%) THEN 3770'CONTROLLA
3790 R%=P%(CG%)+CO%-1:GOSUB 10500:GOSUB 13000:
     PUNCO1%=R%'                                LEGGI
3800 C=7:LOCATE V%(C),H%(C),0:PRINT DEN$


3810 REM C=8:SC%=VAL(IN$(5))'CHIEDI  SOTTOCONTO
3820 REM                          LEGGI E ASSEGNA
3830 REM C=9:LOCATE V%(C),H%(C),0:PRINT DEN$

3840 C=11:GOSUB 11200:CG%=VAL(IN$(5))' CHIEDI IL
                                  CONTO DI GRUPPO
3850 IF CG%<1 OR CG%>NG% THEN 3730'  CONTROLLALO
3860 R%=CG%:GOSUB 10500:GOSUB 13000'      LEGGILO
3870 C=12:LOCATE V%(C),H%(C),0:PRINT DEN$

3880 C=13:GOSUB 11200:CO%=VAL(IN$(5))'     CHIEDI
                                 IL CONTO DI MASTRO
```

```
3890 IF CO%<0 OR CO%>N%(CG%) THEN 3880'CONTROLLA
3900 R%=P%(CG%)+CO%-1:GOSUB 10500:GOSUB 13000:
     PUNCO2%=R%'                    LEGGILO E ASSEGNA
3910 C=14:LOCATE V%(C),H%(C),0:PRINT DEN$
```

16.3.4

Si presenta anche il programma di lettura ed eventualmente di stampa o visualizzazione dell'INDICE che non ha caratteristiche degne di nota particolare salvo quella relativa alla sua importanza primaria rispetto agli altri programmi di cui consente la esecuzione. Infatti i principali valori dei puntatori uttilizzat1 nei programmi stessi vengono assegnati per lettura del file di INDICE così come vengono definite dall'INDICE le chiavi di accesso ai record dei conti.

La codifica è la seguente:

LISTATO 16.5 bis

```
3920 REM C=15:SC%=VAL(IN$(5))'CHIEDI  SOTTOCONTO
3930 REM                                  LEGGILO
3940 C=9:GOSUB 11200:AN$=IN$(5):REM C=16:LOCATE
     V%(C),H%(C),0:PRINT DEN$

3950 CLS:GOSUB 11700:GOSUB 19000'  PRENDI I DATI
                                       DI STAMPA

3960 FOR CR%=PUNCO1% TO PUNCO2%'          RIPETI
                        PER TUTTI I CONTI DA   A

3970 GOSUB 11700:NF%=2:R%=CR%:GOSUB 10500:
     GOSUB 13000'                     LEGGI CONTI
3972 VRIG#(1)=PGR%:VRIG#(2)=NSC%:VRIG#(3)=PSC%:
     VRIG$(4)=DEN$:VRIG#(5)=PRI%
3974 VRIG#(6)=PUL%:VRIG#(7)=D#:VRIG#(8)=A#'
                                ASSEGNA PER STAMPA
3976 GOSUB 19000:GOSUB 11800'             STAMPA
3980 IF PRI%>0 THEN NXT%=PRI% ELSE
3990 GOSUB 18500'REM   STAMPA RIGHE DI MOVIMENTO

4000 WHILE NXT%=<PUL% AND NXT%>0'MENTRE ESISTONO
                      MOVIMENTI DA LEGGERE     (1)

4005 GOSUB 11700
4010 NF%=3:R%=NXT%:GOSUB 10500:GOSUB 13600'
           A PARTIRE DAL PRIMO AL PROSSIMO LEGGI
```

```
4020 REM SE NON VUOI L'ORDINALE CONVERTI IN DATA
     CON LA SUBROUTINE DI 10300
4030 REM SE NON VUOI L'ORDINALE CONVERTI IN DATA
4040 VRIG#(1)=NRE%:VRIG#(2)=DREG:VREG#(3)=NDO%:
     VRIG#(4)=DDOC'            ASSEGNA PER STAMPA
4050 VRIG$(5)= DEN$:VRIG$(6)=DESC$:TIT$=""
4060 IF IM#>0 THEN VRIG#(7)=IM#:VRIG#(8)=0 ELSE
     VRIG#(7)=0:VRIG#(8)=IM#
4070 GOSUB 11800' PRENDI DATI RIGA E STAMPA RIGA
4080 TOTD#=TOTD#+VRIG#(7):TOTA#=TOTA#+VRIG#(8)'
                                   CALCOLA TOTALI

4090 WEND'                        RIPRENDI DA (1)

4095 GOSUB 11985

4100 NEXT CR%'      AL PROSSIMO  CONTO

4110 RETURN
```

CAPITOLO 17

LA CHIUSURA DEI CONTI E IL CALCOLO DEL REDDITO DI ESERCIZIO

17.1 Le fasi di svolgimento delle operazioni di CHIUSURA.
17.2 Il menù di CHIUSURA.
17.3 La CHIUSURA dei conti.
17.4 Il programma.

17.1 Le fasi di svolgimento delle operazioni di chiusura.

Le operazioni di chiusura si svolgono solitamente nelle fasi seguenti:

- lettura dei conti e stampa di un BILANCIO DI VERIFICA;
- registrazione delle rettifiche al conto PROFITTI E PERDITE;
- chiusura dei conti di reddito senza ripresa di saldo al conto PROFITTI E PERDITE e stampa dello stesso;
- chiusura dei conti con ripresa di saldo allo STATO PATRIMONIALE e stampa dello stesso.

La lettura dei conti e la stampa del bilancio di verifica viene effettuata su richiesta dell' operatore ricorrendo a programmi già visti precedentemente.
Lo stesso viene fatto per la registrazione delle rettifiche al conto PROFITTI E PERDITE.
La chiusura dei conti di reddito senza ripresa di saldo al conto PROFITTI E PERDITE può essere realizzata automaticamente se il piano dei conti è stato adeguatamente organizzato. Lo stesso vale per la chiusura dei conti con ripresa di saldo al conto STATO PATRIMONIALE.
Le stampe possono essere fatte sia ricorrendo alle già viste stampe dei movimenti di conto sia ricorrendo a subroutine specifiche.

17.2 Il menù di CHIUSURA.

Le operazioni in successione obbligata sono solo quelle relative alla chiusura dei conti, le stampe del BILANCIO DI VERIFICA e dei conti STATO PATRIMONIALE o PROFITTI E PERDITE possono essere effettuate in qualsiasi momento, pertanto si può

presentare all'operatore un menù di scelta tra le diverse stampe e la chiusura dei conti vera e propria. Detto menù può avere la forma seguente:

```
Operazioni di Chiusura

1) Calcolo e stampa Bilancio di verifica

2) Chiusura dei conti

3) Stampa conto Profitti e Perdite

4) Stampa conto Stato Patrimoniale

5) Fine lavoro

Dammi la tua opzione
```

Figura 17.1

17.3. La CHIUSURA dei conti.

La chiusura dei conti richiede la individuazione dei conti di reddito con ripresa di saldo e la loro distinzione dagli altri.

Detta distinzione può essere effettuata già in sede di apertura dei conti inserendo in un campo del record in apertura una indicazione di appartenenza del conto di gruppo all'una o all'altra categoria. Ad esempio detta indicazione può essere costituita dai caratteri "PP" e "SP" laddove PP sta ad indicare che il conto deve essere chiuso in PROFITTI e PERDITE e SP in STATO PATRIMONIALE.

Con altra tecnica per la apertura si utilizza un piano con struttura di conti di gruppo in parte predefinita secondo lo schema seguente:

Conto di gruppo n.

1) ATTIVITA';
2) PASSIVITA';
3) CAPITALE;
4) COSTI (senza ripresa di saldo);
5) RICAVI (senza ripresa di saldo);
6) PROFITTI E PERDITE
7) STATO PATRIMONIALE

Sulla base di detto schema di base è possibile definire a priori il tipo di saldo di ciascun conto e quindi il conto epilogativo di chiusura senza necessità di controlli.

La chiusura dei conti vera e propria richiede le operazioni seguenti:

1) calcolo del saldo dei conti di reddito con ripresa di saldo;

2) registrazione del saldo in GIORNALE con contropartita al conto PROFITTI e PERDITE (file MOVIMENTI);

3) registrazione del saldo di conto calcolato in DARE o in AVERE in modo da annullarne la differenza (file CONTI);

4) registrazione del saldo nel conto PROFITTI E PERDITE (file CONTI);

Normalmente un piano dei conti è strutturato in più livelli pertanto le operazioni di chiusura richiedono prima la chiusura dei sottoconti, poi la chiusura dei conti di livello superiore.

17.4 Il programma.

Il programma principale sarà costituito come il solito da una inizializzazione con dimensionamento, apertura e lettura indice, apertura del file conti e del file movimenti, presentazione del menù e richiamo delle subroutine in opzione all'operatore.

Il programma di chiusura sarà sostanzialmente costituito da due cicli di calcolo e registrazione, uno per tutti i conti senza ripresa di saldo, l'altro per tutti i conti con ripresa di saldo.

Il programma di chiusura vero e proprio è il seguente (listati 17.1, 17.2):

LISTATO 17.1

```
5500 REM SUBROUTINE DI CHIUSURA CONTI SENZA
     RIPRESA DI SALDO

5510 FOR CG%=4 TO NG%-2'    PER TUTTI I CONTI DI
                                          GRUPPO

5520 TA#=0:TD#=0

5530 FOR CO%=1 TO N%(CG%)'
                      PER TUTTI I CONTI DI MASTRO

5540 A#=0:D#=0
5550 NF%=2:R%=P%(CG%)+CO%-1'
                            PREDISPONI IL PUNTATORE
5560 GOSUB 10500:GOSUB 13000'    LEGGI E ASSEGNA
```

```
5570 TD#=TD#+D#:TA#=TA#+A#'              SOMMA IN
                                     TOTALIZZATORI
5580 SALDO #=D#-A#'                CALCOLA IL SALDO
5590 NRE%=NRE%+1'      PREPARATI A REGISTRARE IN
                                          GIORNALE
5600 DESC$=TI$:IM#=SALDO#:LST%=PUL%:PC%+R%'
                                           ASSEGNA

5610 NF%=3:R%=NRE%:GOSUB 13550:GOSUB 10600'
                              REGISTRA IN GIORNALE

5620 NEXT CO%'   VA AL PROSSIMO CONTO DI MASTRO
5630 R%=CG%:NF%=2:GOSUB 10500:GOSUB 13000'
           PER IL CONTO DI GRUPPO LEGGI-ASSEGNA
5640 D#=TD#:A#=TA#:PUL%=NRE%+1'
          INCREMENTA IL NUMERO DI REGISTRAZIONE
5650 GOSUB 12930:GOSUB 10600'        REGISTRA IN
                                          GIORNALE
5660 R%=NG%-1'                   PER IL CONTO P.P.
5670 GOSUB 10500:GOSUB 13000'   LEGGI E ASSEGNA
5680 D#=D#+TD#:A#=A#+TA#:PUL%=NRE%'        SOMMA
5690 GOSUB 12930:GOSUB 10600'  REGISTRA IN P.P.

5700 NEXT CG%'   VA AL PROSSIMO CONTO DI GRUPPO

5710 RETURN
```

LISTATO 17.2

```
5800 REM SUBROUTINE DI RIPORTO A
                               STATO PATRIMONIALE

5810 FOR CG%=1 TO 3'PER TUTTI I CONTI DI GRUPPO

5820 TD#=0:TA#=0

5830 FOR CO%=1 TO N%(CG%)' PER TUTTI I CONTI DI
                                          MASTRO

5840 A#=0:D#=0
5850 NF%=2:R%=P%(CG%)+CO%-1'        PREDISPONI I
                                        PUNTATORI
5860 GOSUB 10500:GOSUB 13000'   LEGGI E ASSEGNA
5870 TD#=TD#+D#:TA#=TA#+A#
5880 SALDO#=D#-A#'                CALCOLA I SALDI
5890 NRE%=NRE%+1'        INCREMENTA IL NUMERO DI
```

```
                        REGISTRAZIONE IN GIORNALE
5900 DESC$=TI$:IM#=SALDO#:LST%=PUL%:PC%=R%'
                                              ASSEGNA
5910 NF%=3:R%=NRE%:GOSUB 13550:GOSUB 10600'
                        CARICA BUFFER E REGISTRA

5920 NEXT CO%'   VA AL PROSSIMO CONTO DI MASTRO

5930 R%=CG%:NF%=2:GOSUB 10500:GOSUB 13000'
                            PER I CONTI DI GRUPPO
5940 D#=TD#:A#=TA#:PUL%=NRE%'    ASSEGNA TOTALI
5950 GOSUB 12930:GOSUB 10600'
                        CARICA BUFFER E REGISTRA
5960 R%=NG%'                           AL CONTO SP
5970 GOSUB 10500:GOSUB 13000'   LEGGI E ASSEGNA
5980 D#=D#+TD#:A#=A#+TA#:PUL%=NRE%'        SOMMA
5990 GOSUB 12930:GOSUB 10600'   CARICA BUFFER E
                                  REGISTRA S.P.

6000 NEXT CG%'   VA AL PROSSIMO CONTO DI GRUPPO

6010 NF%=2:R%=NG%-1'                AL CONTO P.P.
6020 GOSUB 10500:GOSUB 13000'   LEGGI E ASSEGNA
6030 TD#=TD#+D#:TA#=TA#+A#
6040 SALDO#=D#-A#'       CALCOLA UTILE O PERDITA
6050 NRE%=NRE%+1'        INCREMENTA IL NUMERO DI
                        REGISTRAZIONE IN GIORNALE
6060 DESC$=TI$:IM#=SALDO#:LST%=PUL%:PC%=R%
6070 NF%=3:R%=NRE%:GOSUB 13550:GOSUB 10600'
                        CARICA BUFFER E REGISTRA
6080 R%=NG%:NF%=2'                VA AL CONTO S.P.
6090 GOSUB 10500:GOSUB 13000'   LEGGI E ASSEGNA
6100 D#=TD#:A#=TA#:PUL%=NRE%'
6110 GOSUB 12930:GOSUB 10600'    REGISTRA UTILE
                                      O PERDITA.

6120 RETURN
```

PARTE II

APPENDICI

CAPITOLO 1

NOZIONI FONDAMENTALI DI PROGRAMMAZIONE IN BASIC

1.1 Il programma.
1.2 La forma delle istruzioni di programma.
1.3 I tipi di dati su cui un programma opera:
    1.3.1 le costanti,
    1.3.2 le variabili.

1.1 Il programma

La soluzione di un problema con un calcolatore richiede di solito la preparazione di un programma in un linguaggio che il calcolatore possa comprendere. Un programma è un insieme di istruzioni che consentono di operare su dati in un ordine definito. Un programma descrive i tipi di dati su cui si deve operare, le azioni da fare su quei dati, l' ordine delle azioni.

In un programma si può operare con costanti e con variabili che a loro volta possono contenere diversi tipi di dati.
I tipi di dati fondamentali possono ritenersi i seguenti:

numerici interi;

numerici reali;

alfabetici.

Altri tipi di dati vengono spesso definiti da specifici linguaggi di programmazione.
Le azioni che il calcolatore deve eseguire sono in genere classificate nel modo seguente:

a) azioni di trasferimento dalla memoria;
b) azioni sui dati:
    b1) operazioni aritmetiche e di concatenamento;
    b2) controlli logici sulle relazioni tra i dati.

c) azioni di controllo della sequenza delle istruzioni:
   c1) per la esecuzione in sequenza;
   c2) per la esecuzione con salti condizionati o non;
   c3) per l'arresto del programma.

1.2 La forma delle istruzioni di programma

Un programma in BASIC è costituito di una o più linee di istruzioni, precedute da un numero di linea che ne stabilisce l'ordine di esecuzione.Tale ordine può essere alterato con apposite istruzioni di salto.

Ciascuna linea può contenere una o più istruzioni ognuna delle quali deve essere separata dall'altra da un simbolo che di solito è costituito dai due punti (:).

Con un BASIC interpretato da un apposito programma di trasformazione in linguaggio macchina, ciascuna istruzione viene letta dall'interprete in sequenza e fatta eseguire al calcolatore. I BASIC più comuni sono interpretati, ma esistono anche BASIC compilati in cui tutto il programma, e non le singole istruzioni, viene letto e tradotto in linguaggio macchina prima di essere posto in esecuzione.

Il formato generale di una linea di istruzione è il seguente (fig 1A.1):

n.di linea	parola chiave	corpo della istruzione	return
0010	LET	A=B/D	RETURN

Figura 1A.1

Il numero di linea, in macchine a 8 bit , non deve essere superiore a 65535 (2 elevato alla 16-1).

La chiave è una parola riservata non utilizzabile per altri scopi con la quale si indica all'interprete il tipo di operazione, PRINT, READ, WRITE, LET etc., che il calcolatore deve eseguire sul corpo della istruzione.

Il corpo della istruzione accoglie le variabili o le costanti su cui l'operazione indicata dalla parola riservata deve essere eseguita e gli operatori con i quali si indicano i tipi di relazioni tra le variabili suddette.

Le variabili sono nomi di aree di memoria destinati ad accogliere dati o risultati di un programma.

Nel dare il formato delle singole istruzioni saranno rispettate le seguenti convenzioni:

I le parti scritte in lettere maiuscole costituiscono parole riservate. Debbono essere immesse nel calcolatore così come sono state scritte:

II le parti scritte in lettere minuscole racchiuse tra parentesi angolate vanno inserite a cura dell'operatore: <nome variabile>

III le parti racchiuse tra parentesi quadra sono opzionali;

IV tutti i segni di punteggiatura, salvo le parentesi quadre e le parentesi angolate, vanno inseriti da tastiera come è indicato;

V gli elementi racchiusi tra parentesi tonda, seguiti da puntini di sospensione, possono essere ripetuti;

VI le parti separate da una barra verticale (|) mutuamente si escludono: o si immette l'una, o si immette l'altra.

1.3 I tipi di dati su cui un programma opera

1.3.1 Le costanti.

I dati su cui un programma opera sono accolti in aree di memoria appositamente a ciò destinate. Si può trattare di aree di memoria che accolgono valori che variano nel corso dell'esecuzione del programma, parleremo allora di variabili, oppure di aree che accolgono valori che invece rimangono costanti che si diranno appunto costanti.

Possiamo distinguere tra i due seguenti tipi principali di costanti:

costanti stringa;

costanti numeriche.

Le costanti stringa sono sequenze di caratteri racchiuse tra virgolette come nel seguente esempio:

"numero di studenti".

Le costanti numeriche sono costituite da un elemento dell'insieme dei numeri. Siccome abbiamo diversi insiemi di numeri potremo distinguere tra diversi tipi di costanti numeriche. Avremo quindi i tipi seguenti:

costanti intere	numeri interi dell'insieme -32768..+32767
costanti in virgola fissa	numeri reali con punto decimale
costanti in virgola mobile	numeri reali in forma esponenziale
costanti esadecimali o in ottale	numeri in base sedici in base otto.

Tabella 1A.1

La precisione dipende dal tipo di macchina e dal numero di bit del byte del microprocessore. Con macchine impieganti il microprocessore 8088 la precisione è di sette cifre per le costanti in singola precisione, di 17 per le costanti in doppia precisione. La singola o la doppia precisione viene definita, come per le variabili, mediante rispettivamente il simbolo (!) e il simbolo (#).

In BASIC Applesoft non esiste doppia precisione.

I numeri in esadecimale debbono essere preceduti dal simbolo &H, quelli in ottale dal simbolo &O.

1.3.2 Le variabili.

Le variabili sono aree di memoria destinate ad accogliere dati o risultati di un programma, che possono variare nel corso della sua esecuzione. Mentre una costante in BASIC è identificabile solo attraverso il valore che la costituisce, si deve fare riferimento al valore di una variabile ricorrendo ad un nome. I nomi di variabili sono costituiti da almeno una lettera ed eventualmente da qualsiasi sequenza di caratteri. Non sono ammesse come nomi di variabili le parole riservate (vedasi più avanti pag). I nomi possono essere di qualsiasi lunghezza. Di fatto però nel BASIC Applesoft vengono riconosciuti solo i primi due caratteri.

Le variabili sono distinte in due tipi generali (tab. 1A.2):

VARIABILI STRINGA	il nome e' seguito dal carattere ($); sono destinate ad accogliere sequenze di caratteri non direttamente utilizzabili in operazioni aritmetiche;
VARIABILI NUMERICHE	sono destinate ad accogliere numeri; si distinguono in:
VARIABILI INTERE	il loro nome e' seguito dal carattere (%)
VARIABILI IN SINGOLA PRECISIONE.	il loro nome e' seguito dal carattere (!)
VARIABILI IN DOPPIA	il loro nome e' seguito dal carattere (#)

Tabella 1A.2

NOME$ è pertanto il nome di una variabile alfabetica, SALDO% è il nome di una variabile numerica intera, SALDO! è il nome di una varíabile numerica in singola precisione mentre SALDO # può essere il nome di una variabile in doppia precisione.

Una variabile può essere costituita di una o più serie di elementi, può avere una o più dimensioni: abbiamo in tal caso una variabile multipla. E' possibile fare riferimento a ciascuno dei suoi elementi mediante un indice. Ad esempio, con V(5) si fa riferimento al quinto elemento della variabile V con indice. Un nome di una variabile multipla ha tanti indici quante sono le dimensioni della variabile stessa. Ad esempio, V(5,3) fa riferimento all'elemento della quinta riga e della terza colonna della variabile con indice V.

CAPITOLO 2

LE OPERAZIONI CON LE VARIABILI E LE COSTANTI

2.1. Le operazioni sulle variabili numeriche:
2.1.1 aritmetiche;
2.1.2 logiche e di confronto;
2.1.3 applicazioni di funzioni
2.2 Le operazioni sulle variabili alfabetiche:
2.2.1 concatenamento;
2.2.2 determinazione lunghezza e ricerca sottostringa;
2.2.3 conversioni dei tipi di caratteri;
2.2.4 confronto di stringhe

2.1 Le operazioni sulle variabili numeriche

Sui dati numerici, costanti o variabili, è possibile fare operazioni di diverso tipo che possono riassumersi nel modo seguente:

- Operazioni aritmetiche;
- operazioni di confronto e logiche;
- applicazioni di funzioni.

2.1.1

Per fare eseguire dette operazioni all'elaboratore ci si deve servire di operatori. A due variabili si possono applicare più operatori che vengono eseguiti secondo un ordine di precedenza corrispondente, di solito, a quello delle operazioni aritmetiche.

Gli operatori aritmetici, in ordine di precedenza nella esecuzione sono i seguenti:

^ è il segno dell'operatore elevazione a potenza

x y indica la potenza yesima di x

*,/ sono i segni del moltiplicatore e del divisore

+,— sono i segni di somma e sottrazione.

Per cambiare l'ordine di esecuzione delle operazioni si devono usare le parentesi.

2.1.2

Le operazioni di confronto e logiche si effettuano per confrontare due valori tra di loro. Il risultato del confronto, che può essere vero (-1) o falso (0) può essere utilizzato per prendere decisioni sul flusso delle istruzioni di programma da far eseguire al calcolatore.

Gli operatori relazionali sono i seguenti:

OPERATORE	RELAZIONE	ESPRESSIONE
=	Uguaglianza	X=Y
<>	Disuguaglianza	X<>Y
<	Minore di	X<Y
>	Maggiore di	X>Y
<=	Minore o uguale	X<=
>=	Maggiore o uguale	X>=

Tabella 2A.1

Gli operatori logici servono per effettuare verifiche su relazioni multiple. Il risultato è zero se le relazioni sono false, -1 se le relazioni testate sono vere. Anche gli operatori logici servono per modificare, se necessario, la sequenza delle istruzioni da far eseguire alla macchina.

Gli operatori logici principali sono i seguenti:

NOT, AND, OR, XOR .

Se indichiamo con T o con -1 il valore vero di una variabile, e con F o 0 il valore Falso della stessa gli operatori logici daranno le seguenti risposte:

OPERATORE	VARIABILI		ESPRESSIONE
	X	NOT X	
	T	F	
	F	T	
AND			
	X	Y	X AND Y
	T	T	T
	T	F	F
	F	T	F
	F	F	F

segue

Tabella 2A.2

OR

X	Y	X OR Y
T	T	T
T	F	T
F	T	T
F	F	F

XOR

X	Y	X XOR Y
T	T	F
T	F	T
F	T	T
F	F	F

Tabella 2A.2 (seguito)

2.1.3

Gli operatori funzionali consentono di far effettuare alla macchina delle funzioni intrinseche quali i calcoli delle radici, di seno, coseno, cotangente ed altre numerose implementate nel tipo di basic che si sta usando.

Da ricordare sono le funzioni che consentono di utilizzare solo la parte intera di un numero reale, INT(N), le funzioni di arrotondamento, RND(N), le funzioni di determinazione del segno e del valore assoluto rispettivamente SGN(N) e ABS(N).

L'utente può direttamente definire delle funzioni che, operando dei calcoli sulla base di una lista di paramentri, restituiscono un valore risultante.
Definita una data funzione è possibile ottenerne il risultato facendo semplicemente riferimento ad essa.

Il formato della istruzione è il seguente:

DEF FN <nome> [lista di parametri] = <definizione di funzione>

Si definisca con ART l'area di un triangolo. Siano H e B rispettivamente base e altezza; Avremo le seguenti istruzioni:

```
1000 DEF FNART(H,B)=B*H/2
1010 A=FNART(L,M)
```

In BASIC Applesoft è possibile definire funzioni con un solo parametro.
Risulta comodo impiegare dette funzioni soprattutto quando il loro procedimento di calcolo debba essere impiegato più volte nel corso del programma

2.2 Le operazioni sulle variabili alfabetiche

Le principali operazioni sulle costanti e sulle variabili alfabetiche sono le seguenti:

- concatenamento;
- determinazione della lunghezza e ricerca di sottostringhe;
- conversione dei tipi di caratteri;
- confronto di stringhe.

2.2.1

Il concatenamento si effettua tra due stringhe con l'operatore + e dà come risultato una nuova stringa che contiene tutti i caratteri della prima e tutti i caratteri della seconda nella successione in cui si trovano.

Se B$ contiene "Saldo" e C$ contiene "dare" e A$=B$+CA contiene "Saldodare".

2.2.2 Determinazione della lunghezza e ricerca sottostringhe.

Alcuni operatori consentono di manipolare le stringhe determinandone la lunghezza, estraendone una parte a destra, a sinistra al centro.
Gli operatori in questione sono:

OPERATORE	FUNZIONE	ESEMPIO
LEN(VARS)	determina la lunghezza di VARS	L=LEN(A$) L=9
LEFT$(VARS,L)	prende i primi L caratteri a sinistra.	B$=LEFT$(A$,L) se L=5 B$="Saldo"
MID$(VARS,P,L)	prende L caratteri a partire dal P esimo verso destra	C$=MID$(A$,P,L) se P=4 e L=3 C$="dod"
RIGHT$(VARS,L)	prende gli ultimi L caratteri	D$=RIGHT$(A$,L)

Tabella 2A.3

2.2.3 La conversione dei tipi di caratteri.

La conversione dei tipi dei caratteri di una stringa consente di trasformare caratteri numerici in caratteri alfabetici, alfabetici in numerici, caratteri stringa nel loro ordinale indicante la posizione nel codice ASCII e numeri ordinali nel corrispondente carattere

ASCII. Le operazioni di conversione da carattere numerico in carattere alfabetico sono necessarie nel caso in cui si debbano effettuare delle registrazioni di file su disco.

Nel BASIC 80 e MS-DOS sono disponibili i seguenti operatori (vedi tab. 2A.4) :

OPERATORE	FUNZIONE	ESEMPIO
MKI$(I%)	converte in stringa a due caratteri una variabile intera	LSET I$=MKI$(I%)
MKS$(S)	converte in stringa a quattro caratteri una variabile in singola precisione	LSET S$=MKS$(S)
MKD$(D)	converte in stringa a otto caratteri una variabile in doppia precisione.	RSET D$=MKD$(D)

Tabella 2A.4

Negli esempi riportati le funzioni LSET e RSET sono funzioni di allineamento di necessario impiego in fase di carico buffer di registrazione.

In fase di lettura è necessario procedere alla operazione inversa mediante le seguenti istruzione di conversione:

CVI(VARS)	converte in numero intero una stringa di due caratteri	I%=CVI(I$)
CVS(VARS)	converte in numero in singola precisione una stringa di 4 caratteri	S=CVS(S$)
CVD(VARS)	converte in numero in doppia precisione una stringa di 8 caratteri.	D#=CVD(D$)

Tabella 2A.5

In BASIC Applesoft le funzioni di conversione che abbiamo appena descritto non esistono.Si utilizzano al loro posto le funzioni seguenti (vedi tab. 2A.6):

VAL(VARS)	restituisce il valore numerico di una variabile stringadel tipo "123.."	N=VAL(N$)
STR$(N)	converte la variabile numerica N in una stringa di tipo "12."	A$=STR$(N)

Tabella 2A.6

Dette funzioni esistono anche in BASIC 80 e MS

Altre istruzioni di conversione importanti sono quelle che danno il carattere ASCII corrispondente ad un dato numero ordinale del set 1..256, o 1..128 per macchine a 8 bit (vedasi tab. 2A.7)

CHR$(I%)	restituisce il carattere I esimo del codice ASCII	A$=CHR$(I%) se I% e' 66 A$=B
ASC(VARS)	restituisce il numero di ordine nel codice ASCII	I%=ASC(A$) se A$ e'"b"I% e'66

Tabella 2A.7

2.2.4 Il confronto tra stringhe.

L'impiego di operatori di confronto consente di paragonare stringhe di caratteri definendone l'uguaglianza o la diversitàe in quest'ultimo caso l'ordine alfabetico. La macchina opera confrontando uno per uno il numero d'ordine ASCII dei caratteri delle stringhe.

CAPITOLO 3

IL CONTROLLO DELLA SEQUENZA DELLE ISTRUZIONI DI PROGRAMMA

3.1 Le istruzioni di modifica della sequenza di esecuzione di un programma.
3.2 Le istruzioni di salto incondizionato.
3.3 Le istruzioni di salto condizionato.
3.4 Le istruzioni di Loop.
3.4.1 Istruzioni di ripetizione enumerata (FOR ... NEXT)
3.4.2 Iterazioni condizionate WHILE ... END
3.4.3 Istruzioni Repeat ... Until

3.1 Le istruzioni di modifica della sequenza di esecuzione di un programma

Nel linguaggio BASIC un programma viene eseguito linea per linea nell'ordine del numero di ciascuna di esse.

E' possibile alterare la sequenza di esecuzione con istruzioni di salto. E' possibile far eseguire questi salti incondizionatamente o al verificarsi di date condizioni; è inoltre possibile far eseguire iterativamente gruppi definiti di istruzioni. Possiamo pertanto distinguere tra le seguenti classi di istruzioni di salto:

istruzioni di salto incondizionato
istruzioni di salto condizionato
istruzioni di loop

3.2 Le istruzioni di salto incondizionato.

GOTO <numero di linea>

Con questa istruzione il programma passa alla esecuzione della linea con numero corrispondente a quello indicato nella istruzione di GOTO.
Esempio:

```
10...
20 GOTO 50--|
30 ....     |
40 ...      |
50 PRINT----|
60 ...
```

Figura 3A.1

Nell'esempio il programma esegue la istruzione di linea 10, quindi quella di linea 20 per saltare direttamente a quella di linea 50.

ON <espressione aritmetica> GOTO <lista di numeri di linea>

Con questa istruzione si salta alla linea il cui numero d'ordine corrisponde al valore della espressione aritmetica o della variabile che segue ON. Se il valore della espressione è zero o maggiore del numero dei numeri di linea, il programma prosegue con la istruzione successiva a quella di ON..GOTO. Se il valore è minore di zero la macchina segnala un errore.

Esempio:

160 ON A GOTO 200,300,400.

Sulla base del valore di A la macchina prosegue con la esecuzione della linea 170, se il valore di A è zero, 200, se il valore di A è 1, 300, se il valore di A è 2, 400 se è 3.

Un programma è di solito composto di una parte principale (main program) e di gruppi di istruzioni con funzioni particolari, il controllo dell' input, un calcolo particolare, la stampa di risultati, chiamate subroutine o sottoprogrammi.

Nel programma principale si provvede a chiamare la esecuzione delle subroutine nell'ordine opportuno mediante apposite istruzioni di GOSUB. Le istruzioni di una subroutine chiamata vengono immediatamente eseguite fino a quando non si incontri la istruzione RETURN che restituisce il controllo al programma chiamante per eseguire la istruzione successiva a quella di GOSUB. Un sottoprogramma può essere richiamato un qualsiasi numero di volte; un sottoprogramma può richiamare un altro sottoprogramma con limitazioni legate principalmente al tipo di macchina. E' opportuno servirsi di subroutine almeno nei casi in cui una parte di programma debba essere eseguita più volte.

E' opportuno che i sottoprogrammi appaiano o in testa o in coda al programma principale e non confusi con esso.

Per impedire l'ingresso in un sottoprogramma non voluto e non chiamato si impiegano o istruzioni di salto o istruzioni di END o STOP che interrompano la esecuzione del programma come risulta nell'esempio seguente:

```
            10 REM MAIN PROGRAM
            20 GOSUB 10000'DIMENSIONAMENTO-----|
      |-----30 GOSUB 10100'RICHIESTA DATA -----|---
|----|-----40....                              |   |
|    |   ......                                |   |
|    |                                         |   |
|    |    100 END                              |   |
|    |                                         |   |
|    |   .....                                 |   |
|    |   .....                                 |   |
|    |  10000 REM SUBROUTINE DI DIMENSIONAMENTO  |
|    |  10010 DIM ....                           |
|    |  10020 DIM                                |
|    |   ....                                    |
|    |  10090 RETURN                             |
|    |--------------|                            |
|       10100 REM SUBROUTINE RICHIESTA DATA------|
|       10110 ...
|       .....
|       10190 RETURN
|-------------------|
```

Figura 3A.2

Nel caso vi siano più sottoprogrammi alternativi da far eseguire sulla base del valore di una data variabile o di una data espressione numerica può essere comodo ricorrere alla istruzione ON.. GOSUB.

Il formato della istruzione è il seguente:

ON <espressione> GOSUB <lista di numeri di linea>

Lo scopo di questa istruzione è quello di richiamare una subroutine, che comincia con un dato numero di linea, sulla base del valore assunto dalla espressione tra parentesi angolari.

Se detto valore è zero o maggiore del numero degli elementi della lista dei numeri di linea, il programma esegue la linea successiva a quella di chiamata delle subroutine; altrimenti, se il valore è 1, richiama la prima subroutine della lista, se è 2 esegue la seconda e così via.

Esempio

```
1000 REM OPZIONI
1010 PRINT 1,"SITUAZIONE ARTICOLO"
1020 PRINT 2,"AGGIUNGI ALLO STOCK"
1030 PRINT 3,"TOGLI DALLO STOCK"
1040 PRINT 4,"LIVELLI DI RIORDINO"
1050 PRINT 5,"FINE LAVORO"
1060 AN=VAL(INPUT))
1070 ON AN GOSUB 2000,3000,4000,5000
1080 END
```

Un'altra istruzione importante è quella per cui si richiama un programma totalemente staccato da quello in esecuzione, eventualmente in un disco diverso da quello attualmente in linea. Logicamente si deve chiedere all'operatore di effettuare i cambi necessari alla prosecuzione del programma. La istruzione in esame è la istruzione CHAIN che ha il seguente formato di base:

CHAIN <nome programma> [numero di linea]

Il numero di linea, opzionale, consente di far partire la esecuzione del nuovo programma dal numero di linea indicato.
Esempio:

```
120 CHAIN "PROG1",1000
```

La esecuzione di PROG1 inizierà a partire dalla linea 1000.

3.3 Le istruzioni di salto condizionato

Con le istruzioni di salto condizionato è possibile effettuare salti a linee date all'accertamento di uno o più condizioni. Dette condizioni riguardano la verità di un dato tipo di relazione tra variabili o espressioni. Dette condizioni vengono accertate mediante gli operatori relazionali o quelli logici.

IF <espressione> THEN <istruzione:istruzione:..> | <numero di linea>

```
170 IF A>B THEN PRINT "A È MAGGIORE DI B": PRINT "A="; A; "B="; B
ELSE PRINT"B E' MAGGIORE DI A":PRINT"B=";B;"A=";A
```

Con questa istruzione, se è vera la espressione tra parentesi angolari, il programma passa alla esecuzione delle istruzioni o della linea indicata dopo la parola THEN, altrimenti esegue immediatamente la istruzione della linea seguente.
In BASIC MS, a differenza del BASIC Applesoft, si ha il seguente formato di istruzione

IF <espressione> THEN <istruzione(i)> | <numero di linea> ELSE <istruzione(i)> | <numero di linea>

3.4 Le istruzioni di LOOP

La ripetizione della esecuzione di un dato insieme di linee di programma può essere richiesta mediante apposite istruzioni di ripetizione o di LOOP (ciclo). Abbiamo istruzioni che consentono di ripetere un insieme di istruzioni un numero predefinito di volte, ed altre che consentono di effettuare la ripetizione fino a quando è vera una data condizione o fino a che una data condizione non diventi vera.
Possiamo pertanto distinguere tra i seguenti tre tipi di istruzione di LOOP:

Istruzioni di ripetizione enumerata (FOR...NEXT)

Istruzione di ripetizione condizionata dalla verità di una condizione (WHILE...WEND)

Istruzioni di ripetizione condizionata dalla falsità di una condizione (REPEAT...UNTIL).

Si ricorda che, dato un qualsiasi algoritmo espresso nel linguaggio a salti, esiste un algoritmo, ad esso funzionalmente equivalente, esprimibile esclusivamente attraverso le strutture di sequenza, le strutture IF THEN... ELSE, WHILE...WEND e REPEAT.... UNTIL. Dette strutture possono essere rappresentate dalle seguenti figure in cui il rombo indica una "scelta" il rettangolo operazioni di elaborazioni (vedi fig. 3a.3).

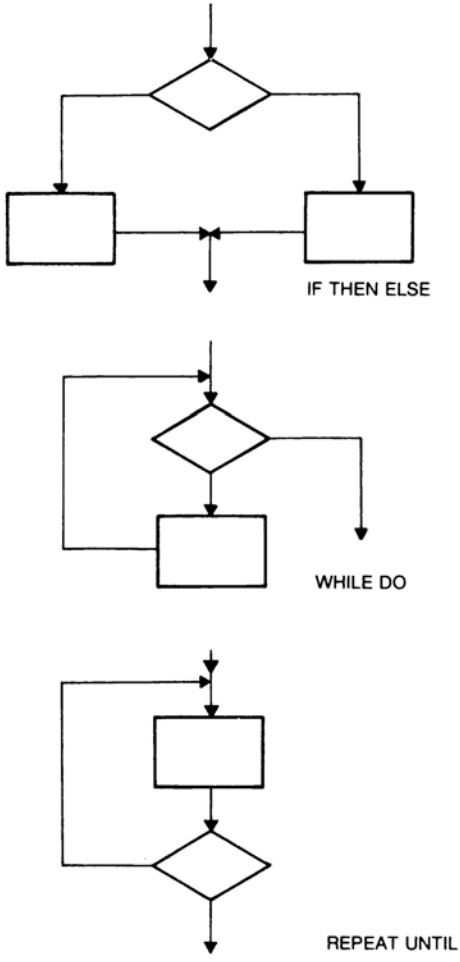


Figura 3A.3

3.4.1 Istruzioni di ripetizione enumerata (FOR ... NEXT).

Le istruzioni di ripetizione enumerata si servono delle parole FOR... NEXT. Il formato delle istruzioni è:

NI..FOR <variabilea> =X TO Y [STEP Z]
nI...
nI...
nI... parte ricorrente
nI NEXT <variabilea>

X,Y,Z sono espressioni numeriche. La variabile messa tra parentesi angolari svolge la funzione di controllo e di contatore del ciclo. Con queste istruzioni si effettua la ripetizione delle istruzioni tra la FOR e la NEXT per un numero di volte pari a (Y+1-X)/Z.

Se la quantità intera Z dello STEP è positiva, Y deve essere intero e maggiore di X; al contrario, se la quantità Z è negativa, X deve essere maggiore di Y.

Alla istruzione NEXT, X viene incrementata di Z, quindi viene effettuato il confronto tra X e Y. Se con un passo positivo X è minore di Y si ripetono le istruzioni a partire da FOR, altrimenti si continua dalla istruzione successiva a quella in cui si trova la parola NEXT.

Le iterazioni possono nidificarsi all'interno di altri cicli FOR... NEXT; non possono però sovrapporsi o incrociarsi (tab 3A.1).

CORRETTI	ERRATI
10 FOR A= X TO Y	10 FOR A= X TO Y
20 FOR B= X TO Y	20 FOR B= X TO Y
30 FOR C= X TO Y	30 FOR C= X TO Y
40 Z=Z+1	40 Z=Z+1
50 NEXT C	50 NEXT B
60 NEXT B	60 NEXT C
70 NEXT A	70 NEXT A

Tabella 3A.1

3.4.2 Iterazioni condizionate WHILE ... WEND.

NI. WHILE <condizione>
nI.... parte ricorrente
nI....
nI.... modifica della condizione
nI WEND

Le istruzioni tra le parole WHILE... WEND sono eseguite finchè è vera la condizione di controllo del ciclo. Detta condizione viene verificata prima dell'inizio del ciclo per cui può darsi che l'insieme delle istruzioni tra le parole WHILE e WEND non venga eseguito nemmeno una volta.

Le istruzioni di WHILE... WEND possono essere nidificate come le istruzioni di FOR .. NEXT.

Le istruzioni di WHILE... WEND non esistono nel BASIC Applesoft, ma possono essere simulate mediante istruzioni di IF.. THEN e GOTO (vedi tab. 3A.2)

```
200 WHILE A<B                          200 IF NOT A<B THEN 240
210 PRINT A                            210 PRINT A
220 A=A+1                              220 A=A+1
230 WEND                               230 GOTO 200
240 PRINT"fine ciclo"                  240 PRINT"fine ciclo"
```

Tabella 3A.2

3.4.3 Istruzioni Repeat ... Until.

Dovrebbero esistere istruzioni per consentire la ripetizione di istruzioni fino a che una data condizione non sia falsa. Dette istruzioni dovrebbero essere del tipo REPEAT... UNTIL con verifica della condizione alla fine del ciclo. Attualmente dette istruzioni non sono implementate nè nel BASIC Applesoft nè in quello MS.

Dette istruzioni però potrebbero essere simulate nel modo seguente:

```
NL (REPEAT)                                300 REM INIZIO CICLO
NL  PRINT "A=";                  parte     310 PRINT "A=";
NL  PRINT A                      ricor=    320 PRINT A
NL  A=A+1 'modifica condizione   rente     330 A=A+1
NL  (UNTIL A=B)                            340 IF NOT (A=B) THEN 300
NL  PRINT"fine ciclo"                      350 PRINT "fine ciclo"
```

Tabella 3A.3

CAPITOLO 4

LE ISTRUZIONI DI TRASFERIMENTO E ASSEGNAZIONE

4.1 Le istruzioni di assegnazione da programma LET, READ... DATA, RESTORE, DIM, CLEAR.
4.2 Le istruzioni di assegnazione da tastiera e da altra periferica INPUT e GET
4.3 Le istruzioni PRINT e le funzioni di tabulazione.

4.1 Le istruzioni di assegnazione da programma LET, READ...DATA, RESTORE, DIM, CLEAR

Le istruzioni di assegnazione sono quelle istruzioni mediante le quali, da programma, da tastiera o da altra periferica , si assegnano valori alle variabili numeriche o stringa.

Le istruzioni di assegnazione da programma sono:

LET, READ... DATA, CLEAR.

Tra le istruzioni di assegnazione può essere compresa anche la istruzione DIM con cui si predefinisce il numero di elementi massimo di una variabile con indice.

LET <variabile> = espressione

Con questa espressione (il LET può essere omesso) si assegna alla variabile alla sinistra del segno di assegnazione(=) il valore dell'espressione. Tra valore ed espressione vi deve essere compatibilità di tipo. Non è possibile, ad esempio, assegnare ad una variabile stringa un valore numerico e viceversa. L'espressione può essere una relazione tra variabili e o costanti, può anche essere una sola variabile o una costante. Esempio:

```
010 LET A=B+C
```

La istruzione READ...DATA ha il seguente formato:

READ <lista variabili>

Per lista di variabili si intende una successione di nomi di variabili separate da una virgola (,).

Con la istruzione READ si leggono sequenzialemente i valori definit da una istruzione DATA e si assegnano, nell'ordine in cui sono letti, alla lista di variabili.

Esempio:

```
020 READ B,C,A$
```

La istruzione DATA ha il seguente formato:

```
DATA <lista di costanti>
```

Con la istruzione DATA si definiscono le costanti numeriche o alfabetiche da assegnare alle variabili di una istruzione READ. Vi deve essere compatibilità di tipo tra variabili e valori assegnati. Il numero delle variabili lette non può essere superiore al numero delle costanti ad esse assegnate.

Esempio:

```
030 DATA 20,30,"RISULTATO"
```

La istruzione RESTORE ha il seguente formato:

```
RESTORE [ <NUMERO DI LINEA>]
```

Questa istruzione consente di rileggere e riassegnare i valori alle variabili della istruzione READ. Se nell' istruzione è indicato un numero di linea la rilettura e la riassegnazione parte dai dati della linea indicata.

Esempio:

```
040 RESTORE 30
```

Non è possibile in Applesoft indicare il numero di linea di rilettura.

La istruzione DIM ha il seguente formato:

```
DIM <variabile multipla(n.elementi, n.elementi)>
```

Con la istruzione DIM si definisce il numero massimo di elementi per ciascuna dimensione di una variabile multipla. Il numero massimo degli elementi di ciascuna dimensione dipende dalla capacità della macchina.

Esempio:

```
050 DIM A(13,17,20),B(50)
```

La istruzione CLEAR ha il seguente formato:

```
CLEAR <lista di variabili>
```

Con questa istruzione si azzerano le variabili indicate nella lista. Senza la indicazione di variabili CLEAR azzera tutte le variabili.
Esempio:

```
060 CLEAR A,B,A$
```

4.2 Le istruzioni di assegnazione da tastiera e da altra periferica INPUT e GET

Le principali istruzioni di assegnazione da tastiera o da altra periferica, dette anche istruzioni di INPUT, sono le seguenti:

INPUT, LINE INPUT, INPUT$( ), INKEY$, GET.

INPUT [<"stringa di richiesta" >]; < lista di variabili >

Con questa istruzione è possibile introdurre nel calcolatore i dati da assegnare alle variabili di programma. Il calcolatore visualizza la stringa di richiesta,se c'è, con un punto interrogativo. A questo punto è possibile immettere i dati che debbono essere compatibili con le variabili cui nell'ordine, mediante la input, vengono assegnati.

```
es. 070 INPUT"DAMMI BASE E ALTEZZA";B,H
```

Volendo introdurre una variabile alfanumerica lunga fino a 254 caratteri, compresi separatori quali virgole, due punti etc., è necessario utilizzare la istruzione LINE INPUT. In tale istruzione l'indicatore di fine linea è costituito da un ritorno carrello e non da una virgola come nella istruzione INPUT. Detta istruzione non esiste in BASIC Applesoft.

Per richiedere un solo carattere in input, da tastiera o da altro terminale, si utilizzano diverse istruzioni a seconda del tipo di BASIC impiegato.

BASIC 80, MS.BASIC

INPUT$ <n.caratteri >
INKEY$

Esempio:

```
080 A$=INPUT$(1)
090 A$=INKEY$
```

Si fa notare che questa ultima istruzione (090) viene impiegata per la lettura di caratteri in arrivo da terminali e deve essere mantenuta attiva per il periodo di invio dei caratteri stessi.
Con la istruzione di riga 080 si assegnano, sempre ad una variabile stringa, tanti caratteri di lettura quanti sono indicati dal numero tra parentesi.

BASIC Applesoft

GET <variabile>

Con detta istruzione si richiama un carattere da tastiera e si assegna alla variabile racchiusa tra parentesi angolari.

Si nota che detta istruzione, accompagnata da parametri indicanti numero logico di file e numero di record, consente in MS.BASIC e in BASIC 80 di leggere i dati di un record (vedasi a pag. 155).

4.3 Le istruzioni PRINT e le funzioni di tabulazione

La visualizzazione o la stampa dei dati e dei risultati avviene mediante istruzioni PRINT che sono appunto istruzioni di trasferimento di dati dal calcolatore al video oppure ad altra periferica. La istruzione ha il seguente formato:

PRINT <lista di espressioni>

Il termine <lista di espressioni> sta per successione di costanti e o variabili separate da un separatore del tipo virgola o punto e virgola.
Esempio:

```
100 PRINT"BASE";B;"ALTEZZA";H;"AREA=";A
```

I separatori impiegati nelle istruzioni di print sono costituiti dai caratteri (;) e (,).

Separatore (;)

Il separatore (;) fa sì che ciascun elemento della lista segua il precedente senza spazi.

Separatore (,)

Il separatore (,) fa sì che ciascun elemento della lista segua il precedente a 14 spazi di distanza.

Se una PRINT termina con (;) o con (,) la istruzione di PRINT successiva visualizzerà la propria lista di espressioni senza salto riga, continuando la lista precedente alla distanza indicata dal separatore.

Le esigenze di tabulazione sul video dei dati da visualizzare vengono soddisfatte mediante apposite funzioni di tabulazione verticale e orizzontale caratteristiche di ciascun tipo di BASIC impiegato.

In BASIC Applesoft la tabulazione verticale e orizzontale viene effettuata con le istruzioni seguenti:

VTAB (numero di linea)
HTAB (numero di colonna)

Dette istruzioni debbono precedere la istruzione di PRINT.
Il numero di linea non può essere superiore a 24, mentre il numero di colonna deve essere compreso tra 1 e 80 se si dispone di una macchina con espansione a 80 colonne, tra 1 e 39 se si dispone di un Apple standard.

Nel caso si debbano tabulare orizzontalmente più scritte ci si può servire, sia in Applesoft che negli altri BASIC, della funzione TAB precedente le scritte da tabulare. Il suo formato è il seguente:

TAB(numero di colonna)

Esempio:

```
110 PRINT ;TAB(10);"importo";TAB(25);"scadenza"
```

Con la istruzione di riga 110 la stringa "importo" viene visualizzata a partire da colonna 10, la stringa "scadenza" a partire dalla colonna 25.

Quest'ultima funzione è disponibile anche nei BASIC MS e nel BASIC 80.

Il posizionamento del cursore in una data posizione verticale e orizzontale del video per una successiva visualizzazione di dati in BASIC MS viene effettuato con la istruzione LOCATE che ha il seguente formato:

LOCATE <riga >, <colonna >, <cursore >

Il numero di riga deve essere dell'insieme 1..24, il numero di colonna deve essere dell'insieme 1..80, il valore del cursore 1 o zero, a seconda lo si voglia visibile o no.

Sempre in MS BASIC è possibile visualizzare dati o scritte all'interno di particolari immagini di stampa mediante la istruzione PRINT USING che ha il seguente formato:

PRINT USING <espressione alfabetica >; <lista di espressioni >

La espressione alfabetica è una costante stringa costituita di particolari caratteri di formattazione. I principali caratteri di formattazione sono i seguenti:

"/nspazi/"

Serve a visualizzare 2+n caratteri di stringa.
Esempio:

```
120 PRINT USING" NOME /    /";NO$
```

Se in NO$ vi è "armando" si otterrà NOME arman

"####.##"

I caratteri di formattazione precedenti servono a visualizzare la parte intera di un numero per tante cifre quanti sono i caratteri che precedono il punto decimale, nel caso tre, ed a visualizzare la parte decimale per tante cifre quanti sono i caratteri di formattazione che seguono il punto decimale.
Esempio:

```
130 PRINT USING"IMPORTO ####.##";I,T
```

Se in I vi è 4000 e in T 365.187 detti valori verranno visualizzati nel modo seguente:

IMPORTO 4000 IMPORTO 365.19

Le istruzioni di PRINT USING non sono disponibili nel BASIC Applesoft.

Impiego di stampanti.
In MS BASIC l'output può essere rivolto alla stampante con l'uso della istruzione LPRINT ed LPRINT USING con formato simile alle già viste istruzioni di PRINT.
Con un Apple si deve segnalare al calcolatore la presenza della stampante con una POKE (POKE 1273,127) ed indicare lo slot di uscita dell'output mediante la istruzione PR che ha il seguente formato:

PR# <numero di slot >
es.

```
140 PR#1
150 PRINT" armando"
```

Detto numero di slot serve ad identificare lo slot in cui è inserita la scheda di collegamento con la stampante. Tutto ciò non è necessario con un Apple II C.

CAPITOLO 5

NOZIONI SUGLI ARCHIVI ELETTRONICI

5.1 Gli archivi elettronici: la loro struttura e le modalità di accesso alle informazioni in essi contenute.
5.2 I file sequenziali: la lettura e la registrazione.
5.3 I file ad accesso casuale: la lettura e la registrazione.

5.1 Gli archivi elettronici: la loro struttura e le modalità di accesso alle informazioni in essi contenute

Un insieme di informazioni sui libri di una biblioteca, sui clienti di una azienda, sulle merci di un magazzino, registrato in qualche supporto fisico, costituisce un archivio. Un archivio è, quindi, un insieme di informazioni attinenti un dato oggetto di conoscenza registrato in un supporto. Un archivio registrato in un supporto magnetico al quale è possibile accedere con un elaboratore prende il nome di archivio elettronico o **file**.

Le informazioni elementari sui singoli libri della biblioteca costituiscono un insieme omogeneo di informazioni che prende il nome di registrazione o **record**.

In ciascun record le informazioni elementari vengono registrate in **campi** o addirittura in **sottocampi**. Così, nel record in cui registriamo i dati dei libri di una biblioteca possiamo avere tanti campi diversi, uno per il titolo, uno per l'autore, un'altro per l'editore etc.. La struttura del record può allora essere rappresentata come nella seguente figura:

```
                          File LIBRI
-------------------------------------------------------------------
| Titolo         | Autore         |    Editore     |Data pubblicazione|
|----------------|----------------|----------------|------------------|
 Campi 1          2                3                 4
```

Figura 5A.1

La **chiave** di un record, di solito, è data da una informazione di un dato campo; si tratta del soggetto di riferimento delle informazioni di un record. Così per i dati dei clienti la chiave potrà essere costituita dal cognome delle persone di cui si ricercano le informazioni, per le informazioni sui libri della biblioteca la chiave potrà essere l'autore e così via.

Il numero d'ordine di un record in un file costituisce quello che si chiama **indirizzo**. L'indirizzo può essere registrato in un campo del record, ma detta registrazione è sovrabbondante in quanto l'elaboratore vi provvede automaticamente e autonomamente.

Può essere interessante registrare in campi appositi gli indirizzi dei record che con quello in lettura o registrazione sono collegati. Detto indirizzo prende il nome di **puntatore**.
Un record può allora essere costituito di un campo riservato all'indirizzo, di campi e sottocampi per i dati, di campi riservati a puntatori. Talvolta si ritiene opportuno riservare file appositi solo per indirizzi e puntatori.

Riassumendo diciamo indirizzo il numero d'ordine di un record in un file; diciamo chiave un termine di riferimento dell'insieme delle informazioni in un record, e uno strumento per la loro identificazione; intendiamo per dati le informazioni elementari che il sistema deve elaborare per trasformarle in informazioni utilizzabili da coloro che prendono delle decisioni.

E' evidentemente importante organizzare i dati in modo da consentirne una conveninete e facile utilizzazione.
Il modo in cui i dati sono organizzati prende il nome di **struttura dati**.
I metodi con i quali i dati vengono utilizzati prendono il nome di **metodi di elaborazione**.

Ai file è possibile accedere **sequenzialmente, direttamente** o **per chiave**.

Nei file ad accesso sequenziale la lettura di un dato record può essere fatta solo dopo che sono stati letti tutti i record che lo precedono.

Nei file ad accesso diretto è possibile accedere ad un dato record dandone il numero d'ordine di registrazione.

Nei file con chiave l'accesso ad un dato record è possibile dandone una chiave.Ad esempio, in un file che contenga degli indirizzi è possibile ottenere l'indirizzo di una persona specificandone il nome.

5.2 I file sequenziali: la lettura e la registrazione

La lettura o la registrazione di un file sequenziale richiedono la seguente successione di operazioni:

I apertura e assegnazione di un nome al file

II definizione del modo di accesso (lettura o registrazione)

III invio o assegnazione dati

IV chiusura file

La apertura del file predispone un'area di lavoro in memoria centrale per le operazioni di lettura e scrittura.
Oltre al nome, in MS-BASIC e in BASIC 80, in apertura viene dato al file un numero logico.
In questa fase viene definito anche il tipo di accesso mediante una stringa contenente la lettera "I", per il modo lettura, e "O", per il modo scrittura.
Pertanto avremo i seguenti schemi di istruzioni di registrazione:

```
nn10 OPEN "O", # numero logico, "NOMEFILE"
nn20 PRINT # numero logico, X$
nn30 CLOSE # numero logico
```

Per la lettura avremo, al posto di OPEN "O" OPEN "I" e al posto della istruzione PRINT, la seguente istruzione:

```
nn20 INPUT # numero logico, X$
```

In BASIC Applesoft si deve in primo luogo assegnare ad una variabile (ad esempio D$) il carattere che segnala alla macchina che le operazioni successive sono da farsi con il disk-driver. Nel caso la istruzione sarà:

```
nnnn D$=CHR$(4)
```

Quindi i comandi successivi di apertura, lettura, scrittura e chiusura si invieranno con delle PRINT D$.
La apertura si farà inviando i comandi con la seguente istruzione:

```
nnnn PRINT D$;" OPEN NOME FILE"
```

Il modo di accesso viene definito con la istruzione:

```
nnnn PRINT D$;" WRITE NOME FILE"
```

oppure con:

```
nnnn PRINT D$;"READ NOME FILE"
```

L'invio dei dati di una stringa X$ viene fatto con la istruzione

```
nnnn PRINT X$
```

la assegnazione dei dati alla variabile X$ vien fatta con la istruzione

```
nnnn INPUT X$
```

la chiusura vien fatta con la istruzione seguente:

```
nnnn PRINT D$;"CLOSE NOME FILE"
```

Lo schema di programma di registrazione è pertanto il seguente:

```
nnnn D$=CHR$(4)
nn10 PRINT D$; "OPEN NOME FILE"
nn20 PRINT D$; "WRITE NOME FILE"
nn30 PRINT X$
nn40
nn90 PRINT D$;"CLOSE NOME FILE"
```

Per la lettura le righe nn20 e nn30 si cambiano nel modo seguente:

```
nn20 PRINT D$;"READ NOME FILE"
nn30 INPUT X$
```

5.3 I file ad accesso casuale: la lettura e la registrazione

La lettura e la scrittura di un file ad accesso casuale richiede la seguente successione di operazioni:

I apertura e definizione lunghezza record
II attribuzione nome e lunghezza campi
III carico buffer e registrazione oppure lettura e conversione
IV chiusura

In MS-BASIC e in BASIC 80 le istruzioni di apertura comprendono anche la assegnazione del nome di un numero logico di file. La definizione del modo di accesso è implicita nella istruzione GET e PUT. Si ricorre pertanto al seguente schema di istruzioni per la registrazioni:

```
nn10 OPEN "R", #numlogico, "NOME FILE", lunghezza record
nn20 FIELD #numlogico, lunghezza campo AS NOME VARIABILE, lunghezza
campo, AS NOME VARIABILE etc.
```

per il carico del buffer di registrazione abbiamo le istruzioni seguenti:

```
nn30 NOME VARIABILE DI CAMPO=LSET VARIABILE DI INPUT 1
nn40 NOME VARIABILE DI CAMPO=LSET VARIABILE DI INPUT 2
etc.
```

```
nn50 PUT #numlogico,numero di record
nn60 CLOSE # numero logico
```

Al posto delle righe 30 e 40 avremo per la lettura le istruzioni seguenti:

```
nn30 GET # numerologico, numero di record
nn40 VARIABILE DI ASSEGNAZIONE=VARIABILE DI CAMPO
```

Con variabili numeriche intere, variabili numeriche in singola e doppia precisione è necessario, prima della registrazione, procedere ad una operazione di trasformazione delle variabili stesse in variabili stringa a 2, 4, 8 caratteri.

Se in NI%, NS, ND# abbiamo valori interi, valori in singola e valori in doppia precisione, la conversione in stringa, ad esempio in I$, S$, DO$ richiede le seguenti istruzioni:

```
nnnn I$=MKI$(NI%)
nnnn S$=MKS$(NS)
nnnn DO$=MKD$(ND#)
```

Al contrario la conversione per la lettura del contenuto delle variabili da stringa a numerico avverrà con le seguenti istruzioni:

```
nnnn NI%=CVI(I$)
nnnn NS=CVS(S$)
nnnn ND#=CVD(D$)
```

naturalmente NI%, NS e D# possono avere qualsiasi altro nome.

In BASIC Applesoft la definizione della lunghezza dei campi deve essere gestita da programma mediante le istruzioni stringa LEFT$ e RIGHT$.

Ad esempio se il record X$ di 32 caratteri è composto dei campi C1$ e C2$, rispettivamente di 20 e di 12 caratteri, all'atto della assegnazione da variabili di programma è opportuno procedere al controllo della lunghezza delle variabili di campo.

Ciò si farà assegnando alle variabili di campo rispettivamente i primi 20 caratteri a sinistra e i primi 12 delle variabili di programma.

Si dovranno pertanto far precedere le istruzioni di registrazione dalle seguenti istruzioni di controllo campo e allineamento:

```
nnnn C1$=LEFT$(A$,20)
nn10 C2$=LEFT$(B$,12)
nn80 X$=C1$+C2$
```

In BASIC Applesoft la lettura o la registrazione di un file ad accesso casuale richiede le seguenti istruzioni:

```
nnnn D$=CHR$(4)
nn10 PRINT D$;"OPEN NOMEFILE,L lunghezza record"
nn20 PRINT D$;"WRITE NOMEFILE,R numerorecord"
nn30 PRINT X$
nn40 PRINT D$;"CLOSE NOMEFILE"
```

Nota bene

Quando un programma Basic incontra degli errori ne deposita il numero di identificazione in una variabile e il numero di linea in cui l'errore si è verificato in un'altra. Per certi tipi di errori il programma si ferma.
Ad evitare che l'esecuzione di un programma si interrompa a causa di un fatto prevedibile (l'inserimento di un dischetto dati diverso da quello richiesto dal programma, la ricerca di un record oltre il numero di quelli registrati ecc.) si creano delle subroutine di gestione degli errori.
Se un programma in MS-BASIC ricerca un file e non lo trova assegna il valore 53 alla variabile ERR e il numero di linea in cui l'errore si è verificato alla variabile ERL.
Con l'istruzione ON ERROR GOTO premessa alla linea in cui si prevede possa verificarsi l'errore si può rinviare il programma alla subroutine di gestione dell'errore e farlo riprendere dopo le correzioni del caso (ad esempio l'inserimento del dischetto giusto).
Le istruzioni di programma e della subroutine di gestione dell'errore potrebbero essere le seguenti:

```
10 ON ERROR GOTO 1000
20 INPUT "NOME FILE";NF$
30 OPEN "I",#1,NF$ / Per aprire un file in lettura se esiste
40 CLOSE 1 / quindi richiuderlo e
50 OPEN "R",#1,NF$ / riaprirlo come file random.

1000 REM SUBROUTINE DI GESTIONE DELL'ERRORE
1010 IF ERR=53 AND ERL=30 THEN PRINT"NON ESISTE":RESUME 20
1020 ON ERROR GOTO O / ALTRI ERRORI INTERROMPONO L'ESECUZIONE
```

CAPITOLO 6

LA STRUTTURA DEGLI ARCHIVI

6.1 La rappresentazione della struttura degli archivi:
6.2 Le tipiche strutture di organizzazione degli archivi:
6.2.1 le strutture sequenziali
6.2.2 le strutture sequenziali con indice
6.2.3 le strutture casuali
6.2.4 le strutture a lista
6.2.5 le strutture ad albero

6.1 La rappresentazione della struttura degli archivi

Gli archivi possono essere costituiti di record della stessa lunghezza aventi la medesima funzione, composti di campi con funzioni specifiche.

La descrizione di un file può allora essere fatta mediante la indicazione del numero di record di cui è composto, dello spazio occupato, della loro struttura in campi e sottocampi.

Gli archivi possono anche essere costituiti di record aventi ciascuno una funzione specifica e allora i file vanno descritti mediante la descrizione delle funzioni svolte dai singoli record.

Nel primo caso i record vanno descritti nel loro contenuto, nei campi che li compongono, nello spazio occupato e nelle variabili utilizzate per la loro registrazione e lettura. Per la descrizione della struttura dei record si usano prospetti detti **"tracciati record"**, per la descrizione delle variabili impiegate nella loro registrazione e lettura si impiegano prospetti detti **"tavole di variabili"**.

Un tracciato record si può presentare con la forma seguente:

```
Nome file _____________

|_____________________________________________________________________
|                 |              |                                    |
| Campo n.1       | campo n.2    |  campo n.3                         | campo
|_________|_____|___|_______|_|_________|_________|________||____
```

Figura 6A.1

Si può impiegare con la stessa funzione il prospetto seguente:

```
APPLICAZIONE _________________

ARCHIVIO ______________

ORGANIZZAZIONE ___________________

LUNGHEZZA RECORD ______

 ____________________________________________________________________
 |                    |              |              |              |     |
 | Descrizione campo  | posizione    | posizione    | n.caratteri  |tipo |
 |                    | iniziale     | finale       |              |     |
 |____________________|____________|____________|______________|_____|
 |                    |              |              |              |     |
 |                    |              |              |              |
 |                                   |
```

Figura 6A.2

In una tavola di variabili si accolgono i nomi delle variabili di campo, i dati sul loro tipo, lunghezza in caratteri, contenuto per cui una tavola può presentarsi con la forma seguente:

```
                         TAVOLA DELLE VARIABILI
 ____________________________________________________________________
 |                  |          |                      |               |
 | Nome variabile   | tipo     |  contenuto           |n.caratteri    |
 |__________________|__________|_______________________|_______________|
 |                  |          |                      |               |
 |                  |          |                      |               |
 |                                                                    |
```

Figura 6A.3

6.2 Le tipiche strutture di organizzazione degli archivi

La struttura di un archivio è data dal modo in cui sono disposti i dati, dai vincoli che si sono stabiliti tra di loro.

Si possono identificare le seguenti strutture semplici fondamentali:

strutture sequenziali;

strutture sequenziali con indice;

strutture casuali;

strutture a lista;

struttura ad albero.

6.2.1

Nelle strutture sequenziali i record sono memorizzati sequenzialmente uno dopo l'altro e possono essere ordinate oppure no:

Struttura sequenziale ordinata

| Rossi | Vioni | Zecchi | Zini | Zoni |

Figura 6A.4

6.2.2

Nelle strutture sequenziali con indice i record sono memorizzati sequenzialmente come in una struttura sequenziale non ordinata, ma viene mantenuta una tabella ordinata di chiavi-puntatori. Detta tabella prende il nome di **master-index**:

Struttura sequenziale con indice

MASTER INDEX

| ROSSI | 2 | VIONI | 5 | ZECCHI | 1 | ZINI | 3 | ZONI | 4 |

MASTER FILE

| 1 | ZECCHI | 2 | ROSSI | 3 | ZINI | 4 | ZONI | 5 | VIONI |

Figura 6A.5

6.2.3
Nelle strutture casuali i record sono registrati nei supporti fisici casualmente con indirizzi determinati con algoritmi operanti, di solito, sulle chiavi.

6.2.4
Nelle strutture a lista i record fisici sono collegati da puntatori al fine di ricavarne la sequenza logica. Di solito si presenta la necessità di identificare il primo record della lista per poter poi accedere ai successivi. detto record prende il nome di **"testa"**. L'ultimo record porta un simbolo di fine lista ammenocchè, trattandosi di una lista circolare, non contenga un puntatore che rimanda alla testa della lista.

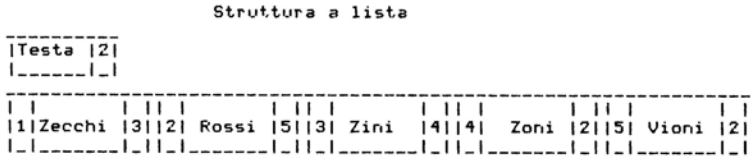


Figura 6A.7

6.2.5
Nelle strutture ad albero da una registrazione primaria, **radice**, dipendono, collegate da puntatori, registrazioni di secondo livello e da queste altre e così via. Si osserva una struttura ramificata con collegamenti gerarchici verticali e non paralleli tra informazioni dello stesso livello.

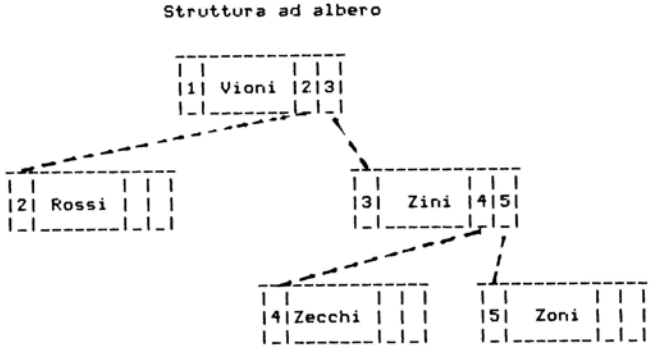


Figura 6A.6

CAPITOLO 7

I METODI DI ELBORAZIONE DELLE STRUTTURE SEQUENZIALI

7.1 I programmi di ordinamento:
7.1.1 Bubblesort,
7.1.2 Subsort,
7.1.3 Subheapsort
7.1.4 Sort di sistema,
7.2 I programmi di ricerca:
7.2.1 la ricerca completa;
7.2.3 la ricerca sequenziale;
7.2.3 la ricerca binaria.
7.3. L'inserimento e la cancellazione di un record

7.1 I programmi di ordinamento

Nelle strutture sequenziali i dati,si è visto, vengono registrati uno dopo l'altro in blocchi contigui.

E' evidente che una struttura in cui l'ordine logico per chiavi corrisponde a quello fisico per indirizzi facilita la ricerca; è quindi comprensibile l'importanza dei programmi di ordinamento o di Sort.

Il primo programma di ordinamento che viene presentato prende il nome di Bubblesort per la sua tecnica che fa emergere gli elementi ordinati come tante bollicine (bubble) di vapore in una pentola di acqua bollente. Il programma Bubblesort ordina gli elementi di una matrice ad una dimensione secondo la procedura seguente:

7.1.1

```
        Bubblesort: ordinamento in senso crescente

INIZIO

        RIPETI

                CONFRONTO tra ciascun elemento ed il suo successivo;

                SCAMBIO


                        dell'elemento che precede con quello che segue,
                        SE il primo è maggiore del secondo;

        PER TUTTI gli elementi -1
```

RICOMINCIA dall'inizio SE vi è stato scambio nella fase 2.

```
15000 REM ********** BUBBLE SORT*****************
15010 INPUT "QUANTI NUMERI DA ORDINARE";N
15020 DIM A(N)
15030 FOR I= 1 TO N
15040 PRINT"DAMMI IL NUMERO";I:INPUT A(I)
15050 NEXT I
```

Con le istruzioni precedenti si dimensiona ad N la variabile multipla A() e si assegnano da tastiera i relativi valori.

```
15060 FLAG=0
```

La variabile FLAG a zero segnala che non vi è stato nessuno spostamento di elementi. Con FLAG a 1 si segnala che almeno uno spostamento è stato effettuato.

```
15070 FOR I = 1 TO N-1
15080 IF A(I)>A(I+1) THEN TEMP= A(I):A(I)=A(I+1):A(I+1)=TEMP:FLAG=1
15090 NEXT I
15100 IF FLAG=1 THEN 15060
```

Per N-1 elementi si fa il CONFRONTO tra ciascun elemento del vettore e il suo successivo. Se il primo elemento di una coppia è superiore al secondo lo si TRASFERISCE nella variabile TEMP, si mette al suo posto il secondo elemento, si TRASFERISCE il primo da TEMP al secondo; si segnala quindi che sono stati effettuati degli spostamenti. Si ricomincia e continua fino a quando FLAG non rimanga zero.

7.1.2

```
]LIST

10000  REM ******************************
10010  REM ******* SUBSORT ***************
10020  REM ******************************
10030  REM  QUESTA SUBROUTINE ORDINA ALFABET
     ICAMENTE PIU' STRINGHE INTRODOTTE COME
     COMPONENTI DEL VETTORE X$(I)
10040  REM  VAR.ESTERNE INPUT:*** X$(I) =***
      VETTORE LE CUI COMPONENTI SONO LE STRI
```

```
     NGHE ALFANUMERICHE DA ORDINARE;*** N =*
     **NUMERO DI STRINGHE DA ORDINARE
10050  REM  VAR. ESTERNE OUTPUT:*** X$(I) =*
     ** VETTORE LE CUI COMPONENTI SONO LE ST
     RINGHE ALFANUMERICHE ORDINATE
10060  REM  VAR. INTERNE:*** MM ***;*** KK *
     **;*** JJ ***;*** II ***;*** LL ***;***
     TT$ ***
10070 MM = N
10080 MM =  INT (MM / 2)
10090  IF MM = 0 THEN 10230
10100 KK = N - MM
10110 JJ = 1
10120 II = JJ
10130 LL = II + MM
10140  IF X$(II) <  = X$(LL) GOTO 10200
10150 TT$ = X$(II)
10160 X$(II) = X$(LL)
10170 X$(LL) = TT$
10180 II = II - MM
10190  IF II >  = 1 GOTO 10130
10200 JJ = JJ + 1
10210  IF JJ > KK THEN 10080
10220  GOTO 10120
10230  RETURN
```

7.1.3

```
]LIST

10000  REM ***********************************

10010  REM ******* SUBHEAPSORT *************

10020  REM ***********************************

10030  REM  *LA ROUTINE ORDINA UN VETTORE IX
     $ DI CHIAVI (ALFANUMERICHE)CONTENENTE N
     X ELEMENTI,MANTENENDO L'ASSOCIAZIONE CO
     N IL CORRISPONDENTE VETTORE IX% DI INDI
     RIZZI(INTERI)
10040  REM  VAR.ESTERNE INPUT :*** NX= ***NU
     MERO DI ELEMENTI DA ORDINARE;*** IX$ **
     *VETTORE LE CUI COMPONENTI SONO LE STRI
     NGHE DA ORDINARE;*** IX% *** VETTORE LE
      CUI COMP. SONO GLI IND. INTERI
10050  REM  VAR.ESTERNE OUTPUT:*** IX$ *** V
```

```
     ETTORE COMPONENTI STRINGHE ORDINATE;***
      IX% *** VETTORE COMP. INDIRIZZI INTERI
      ASSOCIATI
10060  REM  VARIABILI INTERNE:*** L ***;***
     R ***;*** IA$ ***;*** IA% ***;*** J ***
     ;
10070  IF NX = 1 THEN  RETURN
10080 L =  INT (NX / 2) + 1:R = NX
10090  IF L > 1 THEN L = L - 1:IA% = IX%(L):
     IA$ = IX$(L): GOTO 10120
10100 IA% = IX%(R):IA$ = IX$(R):IX%(R) = IX%
     (1):IX$(R) = IX$(1):R = R - 1
10110  IF R = 1 THEN IX%(1) = IA%:IX$(1) = I
     A$: RETURN
10120 J = L
10130 I = J:J = J * 2
10140  IF J > R THEN IX%(I) = IA%:IX$(I) = I
     A$: GOTO 10090
10150  IF J < R THEN  IF IX$(J) < IX$(J + 1)
      THEN J = J + 1
10160  IF IA$ < IX$(J) THEN IX%(I) = IX%(J):
     IX$(I) = IX$(J): GOTO 10130
10170 IX%(I) = IA%:IX$(I) = IA$:IX$(I) = IA$
     : GOTO 10090
10180  RETURN
```

7.1.4

Il sistema operativo MS-DOS offre all'utente la possibilità di impiegare nell'ordinamento dei dati di un file di testo sequenziale il comando SORT.
Il formato di detto comando è il seguente:

SORT [/R] [+n] <file da ordinare [>file ricevente]

L'impiego della opzione /R consente l'ordinamento in senso inverso; l'impiego della opzione +n consente di definire la colonna di dati su cui si deve effettuare l'ordinamento.
Siccome detto comando è un comando di sistema se si è in BASIC si deve uscirne mediante il comando SYSTEM per poterlo utilizzare.

7.2. I programmi di ricerca

In una struttura sequenziale si possono distinguere le seguenti tecniche di ricerca:

ricerca completa;

ricerca sequenziale;

ricerca binaria.

7.2.1

La ricerca completa richiede la lettura e il confronto dei blocchi di registrazione con la chiave cercata fino a che la ricerca non abbia avuto successo o il file non sia stato letto completamente.
Tale metodo di ricerca richiede in media tempi piuttosto lunghi, ma è l'unico possibile con strutture non ordinate.
Tale metodo richiede i seguenti passi:

RIPETI

LETTURA CHIAVE

CONFRONTO del valore della chiave cercata con quella della registrazione appena letta;

SE le due chiavi sono uguali la ricerca ha avuto successo
SE no continua

FINCHE' non hai letto tutto il file.

7.2.2

La ricerca sequenziale si può applicare ad archivi ordinati per chiave.Se le chiavi sono ordinate in modo crescente la ricerca si svolge nelle fasi seguenti:

RIPETI

LETTURA CHIAVE

CONFRONTO del valore della chiave cercata con quella della registrazione appena letta

SE le due chiavi sono uguali la ricerca ha avuto successo;

SE il valore della chiave cercata (anche alfabetica) è inferiore a quello della registrazione la chiave non esiste.

SE il valore della chiave cercata è superiore a quello della registrazione continua

FINCHE' non hai letto tutto il file.

```
nnnn REM RICERCA SEQUENZIALE
nn10 D$=CHR$(4)
nn20 INPUT "DAMMI LA CHIAVE";CH$:L=LEFT$(CH$)
nn30 PRINT D$;"OPEN NOME FILE"
nn40 PRINT D$; "READ NOME FILE"
nn50 INPUT Y$:X$=LEFT$(Y$,L)
nn60 IF CH$>X$ THEN 40
nn70 IF CH$=X$ THEN PRINT"TROVATO":GOTO 90
nn80 PRINT"NON ESISTE"
nn90 PRINT D$;"CLOSE NOME FILE"
n100 END
```

Lo stesso programma in BASIC 80 richiede le seguenti istruzioni:

```
nnnn REM RICERCA SEQUENZIALE
nn10 INPUT "DAMMI LA CHIAVE";CH$:L=LEN(CH$)
nn20 OPEN "I", # numlogico,"NOMEFILE"
nn30 INPUT # numlogico, Y$:X$=LEFT$(Y$,L)
nn40 IF CH$>X$ THEN 30 ELSE IF CH$=X$ THEN PRINT"TROVATO" ELSE
PRINT "NON ESISTE"
nn50 CLOSE # numlogico
nn60 END
```

7.2.3

Se l'archivio è ordinato è possibile procedere alla ricerca con il metodo della ricerca binaria che si realizza nelle fasi seguenti:

RIPETI

DETERMINA:
LA REGISTRAZIONE CENTRALE,
IL LIMITE SUPERIORE,
IL LIMITE INFERIORE

dell'insieme delle registrazioni su cui vuoi effettuare la ricerca;

CONFRONTA la chiave cercata con quella della registrazione centrale;

SE il valore della chiave cercata coincide con quella della registrazione allora la ricerca è conclusa positivamente;

SE il valore della chiave cercata è superiore a quello della registrazione centrale allora
ESCLUDI dalla ricerca tutte le registrazioni che appartengon alla metà con chiave inferiore alla cercata; quindi
CONSIDERA l'altra metà delle registrazioni;

SE il valore della chiave cercata è inferiore a quello della registrazione centrale
ESCLUDI dalla ricerca le registrazioni che appartengono alla metà con chiave superiore e
CONSIDERA l'altra metà.

FINCHE' il limite superiore della parte cercata non sia inferiore al suo limite inferiore

Siano ID$( ) gli elementi alfanumerici di una matrice ordinata in senso crescente.
N sia il numero degli elementi della matrice stessa.
CH$ sia l'elemento da cercare.
LI sia il limite inferiore mobile dell'insieme degli elementi considerati.
LS sia il limite superiore mobile dell'insieme degli elementi.
Sia infine I il puntatore mobile sugli elementi della matrice.

```
nnnn REM RICERCA BINARIA
nn10 LI=1:LS=N
nn20 INPUT"dammi la chiave";CH$
nn30 LI=1:LS=N
nn40 IF LI>LS THEN FLAG=0:GOTO n110
nn50 I=INT((LI+LS)/2)
nn60 IF CH$<ID$(I) THEN LS=I-1:GOTO 40
nn70 IF CH$>ID$(I) THEN LI=I+1:GOTO 50
nn80 FLAG=1:REM TROVATO
n100 PRINT"E' IL NUMERO";I:GOTO n120
n110 PRINT"NON ESISTE"
n120 END
```

7.3 L'inserimento e la cancellazione di un record

La cancellazione di un record in una struttura sequenziale si effettua o mediante la compattazione dei record, oppure mediante registrazione nel record da eliminare di una segnalazione di record eliminato.

L'inserimento di una nuova registrazione richiede lo spostamento in avanti di tutti i record che in ordine logico sequono quello da inserire e quindi l'inserimento vero e proprio.

Queste operazioni sono possibili se i file a struttura sequenziale consentono un accesso diretto. Altrimenti se è ammesso solo l'accesso sequenziale l'inserimento richiede la copia del file fino al record precedente a quello davanti al quale deve essere inserito quello nuovo, la registrazione del nuovo, e la copia degli altri che seguono.

CAPITOLO 8

METODI DI ELABORAZIONE DELLE STRUTTURE A LISTA E CASUALI

8.1 La ricerca e la cancellazione di un record nelle strutture a lista.
8.2. La determinazione di un indirizzo di un record mediante il metodo HASH-CODE.

8.1. La ricerca e la cancellazione di una registrazione nelle strutture a lista

Si fa notare che, oltre alle strutture a lista già viste si impiegano anche altri tipi di liste quali:

LISTE CIRCOLARI (l'ultimo record ha un puntatore al primo della lista);

LISTE BIDIREZIONALI (ogni record ha due puntatori uno per il record che segue l'altro per quello che precede);

PILE (l'accesso, la cancellazione e l'inserimento sono effettuati ad una sola estremità e LIFO)

CODA (l'accesso, la cancellazione e l'inserimento sono effettuati FIFO).

La ricerca di un record in queste strutture si effettua scorrendo la lista fino a quando non si sia trovato il dato cercato.

Le operazioni di inserimento si effettuano correggendo il valore del puntatore del record che precede, in ordine logico, il nuovo da inserire e registrando , come puntatore del nuovo record il puntatore del record precedente.

Le operazioni di cancellazione di un record possono limitarsi alla sola sostituzione del valore del puntatore del record che precede quello cancellato con il valore del puntatore di quest'ultimo.

8.2 La determinazione dell'indirizzo di un record in una struttura casuale mediante il metodo HASH-CODE

In una struttura casuale l'indirizzo di ciascun record si ottiene direttamente dalla chiave; pertanto non esiste relazione tra i singoli record fisici, ma tra questi e le proprie chiavi.

Con il metodo HASH-CODE l'indirizzo di un record è ottenuto applicando un dato algoritmo ai caratteri della chiave o a parte di essi.

Ad esempio, possiamo tener conto del numero d'ordine di ciascun carattere della chiave nel codice ASCII nel modo seguente:

```
nnnn REM        HASH-CODE
nn10 FOR I=1 TO 3:X(I)=ASC(MID$(CH$,I,1)) - 65: NEXT I
nn20 IND=X(1)*26*26+X(2)*26+X(3)-702
nn30 RETURN
```

Con questo algoritmo si possono ottenere 17576 indirizzi pari alle combinazioni con ripetizione di 26 elementi a tre a tre.

PARTE III

PROGRAMMI PER LA GESTIONE DI BANCHE DATI

CAPITOLO 1

DAI PROGRAMMI IN BASIC A QUELLI IN DBASE

1.1 Difficoltà di programmazione in BASIC o in altri linguaggi dello stesso tipo.
1.2 Tipi e vantaggi dei DATA-BASE.

1.1 Difficoltà di programmazione in BASIC o in altri linguaggi dello stesso tipo

La preparazione dei programmi finora presentati ha richiesto venissero affrontati prevalentemente le seguenti difficoltà:

- difficoltà di gestione dei file;
- difficoltà di gestione dei controlli di input output;
- difficoltà di produzione di maschere video per il colloquio con l'operatore;
- difficoltà nella preparazione di programmi di stampa di documenti.

Sia la gestione file che i controlli di input output, le maschere video, i programmi di stampa richiedono lunghe serie di noiose istruzioni e quindi tempo e precisione. Ciò spesso distoglie l'attenzione di chi vuol fare da sè da quelli che sono i problemi fondamentali che con il programma si debbono risolvere e ne fa trascurare la completezza, le possibilità alternative, l'eleganza formale.

Rilevate queste difficoltà alcune case di software hanno prodotto dei programmi che semplificano il lavoro di programmazione per la gestione dei file, il colloquio con la macchina, la stampa di documenti e che prendono il nome di DATABASE.

1.2 Tipi e vantaggi dei DATA-BASE.

Per il tipo di colloquio consentito con l'operatore-programmatore per la richiesta dei loro servizi i DATABASE in commercio possono ricondursi sostanzialmente a due:

- diretti da un sistema rigido di menù;
- diretti da *comandi* costituenti un vero e proprio linguaggio evoluto e generale.

I vantaggi del primo tipo di DATABASE sono quelli di non richiedere all'operatore conoscenze di nessun linguaggio, di esporre con chiarezza mediante una successione di menù le possibilità di sfruttamento del DATABASE stesso, quindi di creare un archivio, di cercare un dato, di preparare e stampare un documento etc.

I vantaggi del secondo sono dati dal fatto che al programmatore viene messo a disposizione un vero e proprio linguaggio di programmazione evoluto e generale, utilizzabile e adattabile alle particolari caratteristiche dei problemi affrontati. Questa adattabilità consente di eseguire immediatamente, o in una successione definità dall'utente, le funzioni desiderate, che invece debbono essere ricercate e messe in esecuzione attraverso il sistema di menù nei DATA-BASE del primo tipo.

Notiamo che se un magazzino può essere gestito tranquillamente con un DATA-BASE diretto da menù le cose si complicano se si devono stabilire dei collegamenti con altre procedure o addirittura gestirle in un sistema integrato. A questo punto diventa inevitabile la scelta per un DATA-BASE del secondo tipo.

Dati gli scopi di questo libro si ritiene opportuno evidenziare le tecniche caratteristiche di programmazione e di impiego di un DATA-BASE programmabile quale il DBII della Ashton-Tate.

CAPITOLO 2

ELEMENTI GRAMMATICALI DI DBII

2.1 I file;
2.2 le variabili di file (di campo);
2.3 le variabili di memoria;
2.4 le costanti;
2.5 i verbi di comando:
    2.5.1 i verbi dei file di dati e di memoria
    2.5.2 i verbi degli strumenti di input-output
    2.5.3 i verbi dei formati video e di stampa
    2.5.4 i verbi dei file di comandi.
2.6 gli operatori:
    2.6.1 aritmetici +, -, *, /, ().
    2.6.2 relazionali <, >, =, , <=, >=, $.
    2.6.3 logici .OR., .AND., .NOT., EOF.
    2.6.4 stringa +
    2.6.5 funzionali aritmetici .INT(espressione numerica) e VAL (STRINGA)
    2.6.6 funzionali stringa LEN, STR, CHR, TRIM.
2.7 i pronomi ALL e NEXT <n>.
2.8 la preposizioni FOR <espressione di relazione>.
2.9 Simboli usati per indicare alcuni elementi grammaticali.

2.1 I file.

DBII opera su file, su variabili di file, su variabili di memoria, su costanti mediante diversi tipi di operatori e di comandi.

I file sono costituiti da successioni di informazioni. Il diverso tipo di impiego nell'ambito di DBII ne consente la seguente classificazione:

- File di dati	.DBF
- File di indici	.NDX
- File di memoria	.MEM
- File di formati	.FMT
- File di rapporti	.FRM
- File di comandi	.CMD

Tabella D2.1

A detti file l'operatore può dare il nome che vuole purchè composto di non più di 8 caratteri. Il programma, in rapporto alla funzione per cui i file sono creati, accetta il nome dato dall'utente e aggiunge una propria estensione preceduta da un punto.

FILE DI DATI .DBF

Sono file che accolgono i dati di base su cui il programma dovrà operare. Hanno un nome dato dall'utente e l'estensione. DBF. Possono avere un massimo di 65535 con un massimo di 1000 byte per record e di 32 campi. Nel DBIi, record passano a 1 miliardo e il numero di byte per record passa da 1000 a 4000 con un massimo di 100 campi per record.

Ogni file di dati ha una intestazione che ne definisce:

- il nome del file
- il numero dei campi
- il nome dei campi
- il tipo dei campi
- la lunghezza dei campi
- il numero dei record.

FILE DI INDICI .NDX

Questi file sono file di chiavi-puntatori ai file di dati.DBF. Servono per facilitare la ricerca dei dati.

FILE DI MEMORIA .MEM

Questi file servono ad accogliere i valori di inizializzazione di variabili di memoria da utilizzare per il deposito temporaneo dei risultati intermedi delle operazioni di DBII. Si possono salvare e richiamare e possono contenere variabili stringa, numeriche e logiche per un massimo di 64.

FILE DI FORMATI .FMT

Sono file di maschere video.

FILE DI RAPPORTI .FRM

Sono file di maschere di stampa documenti.

FILE DI COMANDI .CMD

Sono file di comandi ordinati in modo da costituire un programma. Al comando DO questi file vengono letti e messi in esecuzione. Possono avere nelle macchine a 16 bit l'estensione .PRG.

2.2 Le variabili di campo

Le variabili di campo, come le variabili di memoria, debbono avere un nome di lunghezza non superiore a 10 caratteri costituiti in qualsiasi combinazione purchè non coincidente con i caratteri di un comando e non separati da delimitatori (") e (').

TIPI.
Le variabili possono essere di tre tipi:
- stringa
- numerico
- logico

Il tipo stringa può contenere un massimo di 254 caratteri.

Il tipo numerico può contenere 10 posizioni. Se la variabile viene dichiarata anche come decimale, una posizione viene occupata dal punto decimale.Non possiamo avere numeri decimali con meno di quattro caratteri.

Il tipo logico contiene solo il carattere T (true) o F (false).

2.3 Le variabili di memoria.

Hanno le stesse caratteristiche delle variabili di campo, ma vengono usate solo per il deposito dei risultati intermedi nel corso della esecuzione del programma.Non possono essere impiegate in un numero superiore a 64.

2.4 Le costanti

DBII consente di operare anche con costanti. Si tratta o di numeri fissi all'interno di espressione numeriche, o di successioni di caratteri racchiusi tra virgolette.

2.5. I verbi di comando

Le azioni che il programma deve fare sono comandate da verbi all'imperativo, talvolta sottoposti a condizione. Questi verbi riguardano i file di dati e di memoria e le loro variabili, gli strumenti di input-output, i file di formati e di comandi.

2.5.1
I verbi che riguardano i file di dati e di memoria sono verbi che comandano le azioni seguenti (tab d 2.2):

- uso	- USE, SELECT
- ordinamento	- SORT, INDEX
- posizionamento	- SKIP, LOCATE, GOTO,
- ricerca	- FIND
- visualizzazione	- DISPLAY, LIST, ?.
- sostituzione	- REPLACE
- cancellazione	- DELETE, CHANGE, RELEASE
- salvataggio	- SAVE, COPY, APPEND
- assegnazione e recupero	- RESTORE, STORE.

Tabella D2.2

2.5.2
I verbi che riguardano gli strumenti di input-output, console, stampante, maschere video e maschere di stampanti sono invece (tab d 2.3) :

- uso	- SET PRINT ON-OFF, SET CONSOLE ON-OFF, - SET SCREEN ON-OFF, SET DEFAULT TO <DRIVE>

Tabella D2.3

2.5.3
I verbi che riguardano l'impiego dei formati delle maschere video e di stampa sono (tab d 2.4) :

- uso	- SET FORMAT TO <nome file>
- cancellazione	- ERASE
- visualizzazione	- READ (di tutta la maschera), SAY per
-	- i singoli messaggi.
- input	- GET (per i singoli input)

Tabella D2.4

2.5.4
I verbi che riguardano l'esecuzione di file di comandi sono:

- esecuzione - DO

2.6 Gli operatori

2.6.1 Gli operatori aritmetici:

+ = addizione
— = sottrazione

* = moltiplicazione
/ = divisione
() = raggruppamento tra parentesi.

Questi operatori possono essere utilizzati sia tra costanti, che tra variabili

Es.

```
.? (5+2)*3
.21
.STORE (7+6) TO A
.? A+7
.20
```

? e STORE sono due verbi che significano: VISUALIZZA, ASSEGNA

2.6.2 Gli operatori relazionali: <, >, , =, < + =, > =.

Sono disponibili i seguenti operatori:

```
<   = minore di
>   = maggiore di
=   = uguale a
    = diverso da
<=  = minore o uguale di
>=  = maggiore o uguale di
```

Es.

```
.IF A < 10
.STORE B + I TO
.ENDIF
```

2.6.3 Gli operatori logici: .OR., .AND., .NOT... L'operatore .EOF.

```
.OR.
.AND.
.NOT.
```

Es.

```
.STORE T TO A
.STORE F TO B
.?A.OR.B
.T.

.STORE NOT B TO C
.? A .AND. C
.T.
```

SKIP e GOTO sono verbi che significano rispettivamente: SPOSTATI IN AVANTI DI 1 e VA A.

L'operatore EOF consente di accertare se è stata raggiunta la fine del file in uso. Restituisce infatti .F. (false) o .T. (true) a seconda che detta fine sia stata raggiunta o no.

Es.

```
. ? EOF
.F.
. GOTO BOTTOM
. ? EOF
.F.
.SKIP
.? EOF
.T.
```

2.6.4 Gli operatori stringa

La concatenazione si effettua con l'operatore +.

Es.

```
.STORE "TULLIO" TO PNOME
.STORE "ANGELO" to SNOME
.? PNOME+SNOME
TULLIOANGELO
```

2.6.5 Le funzioni aritmetiche: .INT (espressione numerica), VAL(STRINGA)

Il valore intero di una espressione aritmetica viene dato dalla funzione .INT(ESPRESSIONE ARITMETICA)

Es.

```
. ? INT(345.56)
345
```

Il valore intero di caratteri stringa compresi tra ("0"..."9") viene restituito per una successiva utilizzazione numerica dalla funzione VAL(STRINGA).

Es.

```
.STORE VAL("15") TO B
.? B*3
45
```

2.6.6 Le funzioni stringa: LEN, STR, CHR, TRIM

Con la funzione LEN(stringa) si ottiene la lunghezza della stringa tra parentesi.

```
.STORE "TULLIO" TO NOME
.? LEN(NOME)
6
```

Con la funzione STR(espressione numerica,lunghezza [n. decimali] si trasforma in stringa della lunghezza data, opzionalmente con n decimali, una espressione numerica.

```
.? STR(123.456,10,2)
123.45
```

I caratteri visualizzati sono caratteri stringa e non possono essere utilizzati direttamente in operazioni aritmetiche.

Con la funzione CHR(espressione numerica) si ottiene il carattere ASCII equivalente al numero della espressione tra parentesi.

Con la funzione !(STRINGA) si trasformano i caratteri minuscoli in maiuscoli.
E' una funzione importante infatti DBII considera le minuscole come caratteri diversi dalle corrispondenti maiuscole ed è pertanto opportuno trasformare in maiuscole tutte le stringhe da sottoporre a ricerca.

La funzione TRIM(STRINGA) toglie gli spazi in bianco da una stringa.

2.7 I pronomi ALL, NEXT < n >

Nelle operazioni di accesso ai file è possibile specificare il numero dei record su cui si deve operare con l'uso dei pronomi ALL, o NEXT <n>. ALL significa che l'azione del verbo deve avere per oggetto tutti i record del file in uso, mentre NEXT <n> indica che l'azione deve ricadere sui prossimi <n> record.

```
.DISPLAY ALL
.DISPLAY NEXT 5
```

Con il primo dei comandi sopra indiccati si visualizzano tutti i record, con il secondo solo i prossimi 5.
Con

```
.DISPLAY ALL, NOME, CITY
```

si visualizza di tutti i record solo il nome e la città.

2.8 la preposizione FOR <espressione di relazione>

L'azione di un verbo attivo può essere condizionata da espressioni di relazione, o da complementi di limitazione, introdotti dalla preposizione FOR.
Ad esempio in date circostanze può essere necessario produrre un elenco dei clienti

con un saldo a debito superiore a 500.000 lire, stampare un elenco di indirizzi dei clienti di REGGIO EMILIA etc.. Nel comando DISPLAY si userà la preposizione FOR nel modo seguente:

```
.DISPLAY ALL FOR SALDO>500.000
.DISPLAY ALL, COGNOME, INDIR FOR CITY=REGGIO EMILIA.
```

2.9 Simboli usati per indicare alcuni elementi grammaticali

Lo svolgimento dell'argomento imporrà in seguito di usare dei simboli per indicare alcuni elementi graqmmaticali. Per quanto possibile si usano i simboli del manuale di DBII con qualche modifica legata alle caratteristiche della lingua italiana.In particolare si stabilisce che il singolare indichi un oggetto, un campo, una variabile etc., il plurale indichi una lista di oggetti, variabili, nomi etc. separati da una virgola (vedi tab.d 2.5)

SIMBOLI	SIGNIFICATO
< >	Le parentesi angolari indicano nome o dati da inserire a scelta dell'utente
[ ]	Le parentesi quadre indicano un contenuto opzionale
<stringa>	indica il nome di una variabile stringa
<espressione>	indica una espressione
<espressioni>	indica una lista di espressioni separate da una virgola
<campo>	indica un campo
<campi>	indica una lista di campi separati da una virgola
ALL	significa tutti i record di un file
FOR<espressione>	indica tutti i record che corrispondono al la espressione tra parentesi

Tabella D2.5

CAPITOLO 3

LA SINTASSI DI DBII

3.1 La sintassi di DBII.
3.2 Analisi delle proposizioni: le espressioni.
3.3 Analisi delle proposizioni: le proposizioni di comando.

3.1 La sintassi di DBII

DBII, come tutti i linguaggi, segue delle regole per la costruzione delle singole proposizioni (comandi) perchè abbiano un significato definito.

Segue anche delle regole per la costruzione del "periodo" ,in gergo informatico un "programma".

Dovremo quindi individuare queste regole facendo prima una analisi delle proposizioni semplici, espressioni e proposizioni di comando vere e proprie, successivamente una analisi del periodo o delle proposizioni composte che costituiscono i programmi.

3.2 Analisi delle proposizioni: le espressioni

Variabili o costanti legati da un operatore o da una funzione costituiscono una *espressione*. In DBII possiamo avere espressioni numeriche, stringa o logiche, a seconda del tipo di operatore che ne lega gli elementi.

Ciascuna espressione dipende da una proposizione di comando vera e propria e in questa deve essere contenuta.

L'ordine di esecuzione delle operazioni di una espressione dipende come il solito dagli operatori secondo lo schema seguente (tab d 3.1) :

Operatori aritmetici precedenze	Operatori stringa precedenze	Operatori logici precedenze
1) parentesi, funzioni	parentesi funzioni	.NOT.
2) *, /	relazioni	.AND.
3) +, -	+, - (concatenazione	.OR.
4) relazioni		

Tabella D3.1

3.3 Analisi delle proposizioni: le proposizioni di comando

Le proposizioni con cui si danno i comandi all'elaboratore sono generalmente costituite da verbi opzionalmente seguite da complementi di definizione delle azioni da fare; debbono rispettare le regole seguenti:

1 Ogni comando deve iniziare con un verbo messo immediatamente di seguito al punto di prompt di DBII senza spazi.

2 I verbi possono essere abbreviati ad un minimo di quattro caratteri

3 Ogni singola parola di una proposizione deve essere separata dalla altra da uno o più spazi.

4 Le proposizioni non possono superare i 254 caratteri.

5 Ogni proposizione deve essere chiusa da un " ritorno carrello"

Possono essere di due tipi:

il primo è costituito da un predicato, che indica una azione, da un complemento oggetto (eventualmente con attributi), su cui ricade l'azione, talvolta anche da un complemento di limitazione accompagnato dai suoi attributi;

il secondo è costituito da un predicato, da un eventuale complemento oggetto e da un complemento di luogo.

L'oggetto può essere costituito da tutto un archivio (ALL), da uno o più campi di file, dai prossimi 5 record (NEXT 5) etc.

Si considerino le seguenti tipiche frasi di DBII:

.LIST
.USE MACOR
.DISPLAY ALL FOR ZONA="EMILIA"
.DELETE ALL FOR PAGAMENTI=0

Queste frasi possono ricondursi alla forma seguente:

VERBO	COMP. OGGETTO	COMP. DI LIMITAZIONE	ATTRIBUTI
LIST			
USE	MACOR		
DISPLAY	ALL	FOR ZONA	="EMILIA"
DELETE	ALL	FOR PAGAMENTI	=0

Tabella D3.2

Si considerino ora le frasi seguenti:

.SKIP
.SKIP-1
.GOTO TOP
.STORE"MODIFICA"TO MODO
.RESTORE FROM CLIENTI

Queste frasi possono ricondursi alla forma seguente:

VERBO	COMPLEMENTO OGGETTO	COMPLEMENTO DI LUOGO
SKIP		
GOTO		TOP
STORE	"MODIFICA"	TO MODO
RESTORE		FROM CLIENTI

Tabella D3.3

CAPITOLO 4

LA GESTIONE DEI FILE: I COMANDI FONDAMENTALI

4.1 La creazione di un archivio
4.2 La visualizzazione della struttura
4.3 La copia
4.4 La modifica
4.5 La utilizzazione
4.6 La registrazione
4.7 La visualizzazione dei dati
4.8 L'ordinamento
4.9 La ricerca
4.10 La correzione
4.11 La cancellazione.

4.1 La creazione di un archivio

Prima di salvare le informazioni in un archivio è necessario dargli un nome e stabilirne la struttura determinando le caratteristiche dei suoi record dichiarando tipi, lunghezza e nome dei campi.

Queste dichiarazioni si fanno dopo avere dato alla macchina il seguente comando:

.CREATE

dopo di chè appariranno sullo schermo le richieste (video d 4.1) :

```
***  dBASE II/86  Ver 2.3D  29 Nov 82
. CREATE INDIRCLI
ENTER RECORD STRUCTURE AS FOLLOWS:
 FIELD    NAME,TYPE,WIDTH,DECIMAL PLACES
 001      COGNOME,C,20
 002      NOME,C,20
 003      CITY,C,20
 004      PROVINCI,C,4
 005      VIA,C,20
 006      CODPOST,N,6
 007      NTEL,N,9
 008      ATTIVI,L,1
 009      SCONTO,N,4,2
 010
INPUT DATA NOW?
?
?
?
?
?
?
```

Video 4.1

4.2 La visualizzazione della struttura

I file di dati DBII conservano nella loro parte iniziale tutti i dati inerenti la loro struttura e cioè il nome del file, il numero dei campi, il nome di ciascun campo, la lunghezza in caratteri e il tipo.

La visualizzazione della struttura dell'archivio si effettua con i comandi:

.LIST STRUCTURE oppure

.DISPLAY STRUCTURE

DBII oltre a visualizzare la struttura dà la data dell'ultimo aggiornamento e il numero dei record registrati (video d 4.2).

```
. USE INDIRCLI
. DISPLAY STRUCTURE
STRUCTURE FOR FILE:  INDIRCLI.DBF
NUMBER OF RECORDS:   00006
DATE OF LAST UPDATE: 08/20/84
PRIMARY USE DATABASE
FLD        NAME       TYPE WIDTH    DEC
001      COGNOME       C    020
002      NOME          C    020
003      CITY          C    020
004      PROVINCI      C    004
005      VIA           C    020
006      CODPOST       N    006
007      NTEL          N    009
008      ATTIVI        L    001
009      SCONTO        N    004     002
** TOTAL **              00105
.
```

Video 4.2

Con il comando ? si può visualizzare il contenuto di una espressione o di una variabile:

```
.? (3+5)
8
.? NOME
TULLIO
```

4.3 La copia

La copia di un archivio in uso in uno nuovo si può effettuare con il comando:

```
.COPY TO <NOMENUOVO FILE>
```

La copia della struttura di un archivio in uso in un nuovo archivio si effettua con i comandi:

```
.COPY STRUCTURE TO <NOME NUOVO ARCHIVIO>
```

E' anche possibile copiare solo una parte della struttura nel nuovo archivio con:

```
.COPY FIELD <NOMI CAMPO> TO <NOME NUOVO ARCHIVIO>
```

E' anche possibile creare una nuova struttura e copiarvi i dati di un vecchio archivio con:

```
.CREATE <NOME NUOVO ARCIVIO>
.APPEND FROM <NOME VECCHIO ARCHIVIO>
```

4.4 La modifica della struttura

Se su un file in uso è necessario fare dei cambiamenti di struttura si deve impiegare il comando .MODIFY.

Si fa notare che l'impiego di detto comando implica la distruzione dell'archivio in uso per cui è prudente provvedere, prima della sua distruzione, alla sua copia.

4.5 La utilizzazione

L'uso di un archivio richiede la lettura dei dati di struttura dello stesso.

La utilizzazione di un archivio può essere ottenuta con il comando USE NOME FILE come nel seguente esempio:

```
.USE INDIRCLI
```

Dal momento in cui un archivio è in uso DBII riferirà a quello tutti i comandi.
E' possibile utilizzare simultaneamente due archivi di dati. Per fare ciò è necessario richiederne l'uso con la seguente successione di comandi:

.SELECT PRIMARY
.USE INDIRCLI
.SELECT SECONDARY
.USE COGNOME

.SELECT PRIMARY e SELECT SECONDARY riservano zone di memoria distinte al file primario e al file secondario.

4.6 La registrazione

Dal momento in cui un file è stato creato e messo in uso è possibile provvedere alla registrazione dei dati. Per detta registrazione è necessario utilizzare il comando:

.APPEND

dopo avere eseguito questo comando DBII visualizza il numero di record e il nome dei campi da registrare.

Si potranno appendere tutti i record che si vorrà. Per terminare la registrazione alla richiesta di introduzione dei caratteri del primo campo si deve rispondere con "ritorno carrello" <CR>.

4.7 La visualizzazione dei dati

Per la visualizzazione del contenuto di un archivio si possono utilizzare i comandi LIST o DISPLAY. Con LIST si visualizza il contenuto di tutti i record, con DISPLAY solo il contenuto dell'ultimo su cui si è operato a meno che non si accompagni il verbo con un ALL.
Si può limitare la visualizzazione al contenuto di alcuni campi soltanto indicando i nomi dei campi da visualizzare.

E' anche possibile visualizzare solo quei record che hanno uno o più campi che rispondono a date condizioni.
Si vedano gli esempi seguenti (video d 4.3) :

```
. LIST COGNOME, CITY, VIA
00001  TRINCARDI          REGGIO EMILIA        PIETRO NENNI 4
00002  MELEGARI           REGGIO EMILIA        EMILIA S.PIETRO 52
00003  BENEVELLI          REGGIO EMILIA        SANTI N4
00004  NICOLINI           PARMA                DEL BATTISTERO
00005  NORINI             BOLOGNA              UGO BASSI
00006  RINALDINI          PARMA                GRAMSCI
.
. LIST COGNOME, CITY, VIA FOR CITY="REGGIO EMILIA"
00001  TRINCARDI          REGGIO EMILIA        PIETRO NENNI 4
00002  MELEGARI           REGGIO EMILIA        EMILIA S.PIETRO 52
00003  BENEVELLI          REGGIO EMILIA        SANTI N4
.
. DISPLAY COGNOME, CITY, VIA
00006  RINALDINI          PARMA                GRAMSCI
.
. DISPLAY ALL COGNOME, CITY, VIA
00001  TRINCARDI          REGGIO EMILIA        PIETRO NENNI 4
00002  MELEGARI           REGGIO EMILIA        EMILIA S.PIETRO 52
00003  BENEVELLI          REGGIO EMILIA        SANTI N4
00004  NICOLINI           PARMA                DEL BATTISTERO
00005  NORINI             BOLOGNA              UGO BASSI
00006  RINALDINI          PARMA                GRAMSCI
.


. DISPLAY ALL COGNOME, VIA, CITY FOR CITY="PARMA"
00004  NICOLINI           DEL BATTISTERO       PARMA
00006  RINALDINI          GRAMSCI              PARMA
.
```

Video 4.3

4.8 L'ordinamento

Può essere necessario ordinare i dati di un archivio sulla base di un dato campo chiave. Per questo scopo si possono utilizzare due comandi diversi con diversi effetti. Con uno (il comando SORT) si crea un nuovo file ordinato con la estensione .DBF che dovrà essere messo in uso al posto del precedente, con l'altro (comando INDEX) si crea semplicemente un file con l'indice ordinato delle variabili del campo chiave.

Un file indicizzato consente di impiegare il comando FIND che cerca il record che contiene la chiave data.

La creazione del file ordinato si realizza con i comandi SORT e INDEX seguenti:

```
. USE INDIRCLI
.
. SORT ON COGNOME TO COGNOME
SORT COMPLETE
.
. USE COGNOME
.
. LIST COGNOME, NOME, CITY
00001  BENEVELLI          ALFONSO          REGGIO EMILIA
00002  MELEGARI           MARIAPIA         REGGIO EMILIA
00003  NICOLINI           RINO             PARMA
00004  NORINI             PAOLO            BOLOGNA
00005  RINALDINI          GIOBATTA         PARMA
00006  TRINCARDI          TULLIO           REGGIO EMILIA
.
.



. USE COGNOME
.
. INDEX ON CITY TO CITTA
00006 RECORDS INDEXED
.
. LIST COGNOME, NOME, CITY
00004  NORINI             PAOLO            BOLOGNA
00003  NICOLINI           RINO             PARMA
00005  RINALDINI          GIOBATTA         PARMA
00001  BENEVELLI          ALFONSO          REGGIO EMILIA
00002  MELEGARI           MARIAPIA         REGGIO EMILIA
00006  TRINCARDI          TULLIO           REGGIO EMILIA
.
```

Video 4.4

Si nota che il comando .SORT ON COGNOME TO COGNOME ha creato un file di DATA-BASE COGNOME.DBF. Detto file è stato messo in uso prima di provvedere alla stampa ordinata sulla chiave cognome del file stesso.

Successivamente è stato creato con il comando .INDEX ON CITY TO CITTA un file CITTA.NDX che è stato messo in uso con il comando .USE COGNOME INDEX CITTA per ottenere i dati dei record con le città in ordine alfabetico.

4.9 La ricerca

La ricerca di un dato può essere effettuata record per record oppure, in un file con indice, dando la chiave desiderata in una istruzione FIND.

Nella ricerca record per record si può iniziare dal primo (TOP), dall'ultimo (BOTTOM), oppure da un record definito dall'utente.

Il posizionamento su questi record si effettua con le istruzioni seguenti:

.GOTO TOP
.GOTO BOTTOM
.GOTO 5

Quindi si potrà visualizzarne il contenuto con una istruzione DISPLAY.
La ricerca per chiave può essere effettuata su file indicizzati. E' necessario mettere in uso il file dati indicando anche il file di indici con le istruzioni seguenti:

```
.USE COGNOME INDEX CITTA
```

quindi si può provvedere alla ricerca del cliente abitante in una data città con la istruzione:

```
.FIND BOLOGNA
.DISPLAY COGNOME,NOME,VIA
```

4.10 La correzione

Se una data registrazione deve essere modificata si possono utilizzare due tipi di comando, uno che consente di esaminare l'intero record modificandone in tutto o in parte il contenuto, l'altro che sostituisce il contenuto di un dato campo con dati caratteri.
Il primo è il comando .EDIT seguito dal numero di record che si vuole esaminare, il secondo è il comando .REPLACE e richiede di essere preceduto dal numero si record in cui la sostituzione deve essere effettuata. Il loro uso viene qui di seguito esemplificato.
I comandi

```
.GOTO 5
.EDIT
```

consentono di posizionarsi sul record numero 5 quindi di visualizzarne il contenuto e di operare le necessarie rettifiche.

Dopo di che DBII risponderà:

```
ENTER RECORD    ,FIELD (    OR NAME), NEW VALUE COORD:
```

I comandi

```
.GOTO 4
.DISPLAY
.REPLACE ATTIVI WITH F
```

permettono di posizionarsi sul record numero quattro, di visualizzarne il contenuto e di sostituire nel campo ATTIVI il valore logico F a T.
Un aumento del 10% del prezzo di listino di tutti gli articoli in stock si effettuerebbe con questi semplici comandi:

```
.USE MACOR
.REPLACE ALL PREZZOL WITH (PREZZOL*1.10)
```

4.11 La cancellazione

La cancellazione dei dati di un archivio avviene con misura di prudenza effettuato in due tempi successivi,il primo per marcare il record da cancellare con un asterisco in attesa di una decisione successiva, il secondo per la soppressione definitiva.

Posizionati su un dato record con .NRECORD si provvede alla prima esclusione con il comando .DELETE.

Se si desidera la eliminazione definitiva allora si dà il comando .PACK

CAPITOLO 5

I COMANDI PER LA GESTIONE DEI FILE DI VARIABILI DI MEMORIA

5.1 I comandi di inizializzazione e assegnazione: la istruzione STORE
5.2 I comandi di salvataggio;
5.3 I comandi di richiamo;

5.1 I comandi di inizializzazione e assegnazione: la istruzione STORE

Qualsiasi programma richiede l'impiego di variabili di memoria per il calcolo e il temporaneo deposito di dati intermedi. DBII dà la possibilità all'utente di utilizzare un massimo di 64 variabili che possono contenere al più 254 caratteri (racchiusi tra virgolette), o dei numeri a 10 cifre, o i valori logici T (true) F (false).

Anche se è possibile operare direttamente sulle variabili di campo è buona norma servirsi delle variabili di memoria quindi rimpiazzare i risultati nel record o nei record desiderati.

E' comodo provvedere ad inizio programma alla definizione dei nomi e alla assegnazione di valori di inizializzazione (blank o 0) alle variabili che si prevede di utilizzare in seguito. Nei nostri programmi un programma apposito verrà destinato a questa inizializzazione.

La assegnazione e quindi anche la inizializzazione delle variabili di memoria si effettua con il comando STORE. Si vedano i seguenti esempi:

La inizializzazione verrà fatta nel modo seguente:

```
.STORE " " TO NOME
.STORE 0 to PREZZOL
.STORE 0.00 TO IVA
```

La successiva assegnazione di valori nel corso del programma si effettua nel modo seguente:

```
.STORE "CARLO" TO NOME
.STORE 15000 TO PREZZOL
.STORE .18 TO IVA
```

5.2 I comandi di salvataggio

Le variabili di memoria da utilizzare nel corso della esecuzione di un programma possono essere salvate in un file con un nome dato dall'utente con una estensione .MEM data da DBII.

Questo salvataggio, probabilmente con i valori di inizializzazione, richiede la seguente istruzione:

```
.SAVE TO INIZIA
```

DBII conserverà le variabili su disco nel file INIZIA.MEM

5.3 I comandidi richiamo

A inizio programma la inizializzazione delle variabili di memoria può essere fatta richiamando i dati delle variabili salvate con la istruzione RESTORE.

```
.RESTORE FROM INIZIA
```

Volendo visualizzare il contenuto di una variabile di memoria si possono impiegare i seguenti comandi:

```
.DISPLAY NOME
.LIST NOME
```

Volendo visualizzare il contenuto di tutte le variabili si dovrà usare il comando:

```
.LIST MEMORY
```

CAPITOLO 6

COMANDI DI INPUT-OUTPUT

6.1 ACCEPT [" <messaggio>"] TO <VARIABILE>
6.2 INPUT [" <messaggio>]" TO <variabile>
6.3 WAIT TO <variabile>
6.4 SET FORMAT TO [SCREEN]
[PRINT]
[<format file>]
6.5 @ <n. riga, n. colonna> SAY "messaggio"
6.6 @ <n. riga, n. colonna> GET <variabile>
6.7 READ
6.8 Altri importanti comandi SET...

6.1 ACCEPT [" <messaggio>"] TO <VARIABILE>

Questo comando consente di immettere dati da tastiera con assegnazione ad una variabile stringa e un opzionale previo messaggio all'operatore.

```
.ACCEPT "DAMMI IL NOME" TO NOME
DAMMI IL NOME: GINO
. DISPLAY NOME
NOME (C) GINO
```

6.2 INPUT [" <messaggio>]" TO <variabile>

Questo comando consente di immettere dati da tastiera con assegnazione a variabili di qualsiasi tipo. Il tipo viene definito all'atto della immissione;
il tipo stringa si distingue dal numerico racchiudendone i caratteri tra virgolette.
INPUT viene quindi usato per i dati numerici mentre ACCEPT viene impiegato per quelli stringa e logici.

6.3 WAIT TO <variabile>

Chiede in input un solo carattere prima di proseguire quindi, lo assegna alla variabile se esiste la opzione TO.

6.4 SET FORMAT TO [SCREEN]
[PRINT]
[<format file>]

Le prime due forme determinano dove (schermo o stampante) vanno inviati i comandi , l'ultima determina per quale *format-file* va dato il comando di lettura READ.

6.5 @ <n. riga, n. colonna> SAY "MESSAGGIO"> [USING <FORMAT>] <n. riga, n. colonna> SAI <variabile> [USING <FORMAT>]

Preceduto da un SET FORMAT TO consente di visualizzare un messaggio di una videata o una variabile nella posizione data dalle coordinate*3 n. riga, n. colonna*, opzionalmente con il formato definito dalla opzione USING. Al posto di *n. riga, n. colonna* possono essere messi sia valori costanti che nomi di variabili.

Il messaggio viene visualizzato nel momento in cui è dato il comando di lettura video READ.

Si fa notare che le variabili di assegnazione delle *SAY* debbono essere state inizializzate prima della esecuzione dei comandi READ.

La assegnazione va fatta in bianco con tanti spazi quanti sono i caratteri da visualizzare mediante il comando:

STORE" " TO VARIABILE

6.6 @ <n. riga, n. colonna> GET <variabile>

Preceduto da un SET FORMAT TO, consente di chiedere un input da tastiera, eventualmente con il controllo PICTURE <FORMAT> all'atto della esecuzione del comando READ.

Si fa notare che le variabili di assegnazione delle *GET* debbono essere state inizializzate prima della esecuzione dei comandi READ.

La assegnazione va fatta in bianco con tanti spazi quanti sono i caratteri da chiedere in input mediante il comando:

STORE" " TO VARIABILE

6.7 READ

Provvede alla esecuzione di tutti i comandi e alla richieste di input dei comandi GET di un *format-video*.

6.8 Altri importanti comandi SET...

SET TALK ON fa si che i messaggi di DBII siano visualizzati mentre SET TALK OFF li elimina.

SET PRINT ON fa si che l'output sia inviato in eco anche alla stampante.
SET PRINT OFF toglie l' output alla stampante.

SET ECHO ON visualizza per l'azione di *debug* i comandi di un command file.
SET STEP ON provvede alla esecuzione passo-passo per facilitare il *debug*.

CAPITOLO 7

I PROGRAMMI: I FILE DI COMANDI

7.1 I file comandi.
7.2 Come scrivere un file di comandi.
7.3 La esecuzione dei file di comandi.
7.3.1 la esecuzione condizionata: IF...ELSE...ENDIF,
7.3.2 la esecuzione condizionata di un comando tra molti: DO CASE...ENDCASE.
7.3.3 la esecuzione ripetuta: DO WHILE....ENDDO
7.3.4 il richiamo di esecuzione di un altro file.

7.1 I File di comandi

Qualsiasi successione di comandi salvata in un file di nome dato dall'utente con estensione.CMD costituisce un "file di comandi". La sua esecuzione può essere chiamata con il comando DO <nome file.CMD>. Questo comando ordina a DBII di provvedere alla lettura e alla esecuzione dei singoli comandi registrati nel file stesso nella sequenza in cui sono stati registrati.

Svolgono la stessa funzione i file con estensione .PRG.

7.2 Come scrivere un file di comandi: MODIFY COMMAND <nomefile.est>

CTRL-W

Per la registrazione di un file di comandi si può utilizzare o "l'editore" di sistema operativo MS-DOS (CPM) o quello di DBII o addirittura, per sfruttarne le più potenti capacità di trattamento dei testi, l'editore di un WORD-PROCESSOR, WORD-STAR o altro.

Per editare un file di comandi o un file di formati mediante l'"editore" di DBII si deve impiegare il comando:

.MODIFY COMMAND <nome file.estensione>

e quindi si devono registrare i comandi nella sequenza voluta.

Finito il lavoro di "editing" si devono premere i tasti:

CTRL-W.

A questo punto il file è salvato .

7.3 La esecuzione dei file di comandi

I file con estensione .PRG possono essere messi in esecuzione con il comando DO <nome file>.

I file con nome di estensione.CMD possono essere messi in esecuzione mediante il comando:

```
DO <nome file.CMD>
```

I comandi di file vengono messi in esecuzione uno alla volta nell'ordine in cui sono stati registrati, a meno che l'ordine di esecuzione non sia stato alterato da apposite istruzioni.

7.3.1 La esecuzione condizionata: IF ... ELSE ... ENDIF.

La esecuzione di uno o più comandi può essere sottoposta a condizione come nei seguenti esempi:

```
IF ALTRO="NO"
        ERASE
        QUIT
ENDIF

IF FILE(INDIRCLI)
        USE INDIRCLI
        ELSE
        ?"FILE NON TROVATO"
ENDIF
```

Nel primo caso se ALTRO è uguale a "NO" la macchina cancella lo schermo e abbandona DBII.

Nel secondo accertata la esistenza del file "INDIRCLI" lo mette in uso oppure visualizza "FILE NON TROVATO".

I comandi condizionati debbono essere dati secondo la seguente forma generale:

```
IF CONDIZIONE
        COMANDO
ENDIF
```

oppure :

```
IF CONDIZIONE
      COMANDO 1
ELSE
      COMANDO 2
ENDIF
```

Sono ammesse anche strutture più complesse con successive serie di ELSE IF chiuse alla fine dai corrispondenti ENDIF.

7.3.2 La esecuzione condizionata di un comando tra molti: DO CASE ... ENDCASE.

Quando si deve scegliere la esecuzione di uno o più comandi tra molti possibili si può impiegare la istruzione "esegui in caso di", particolarmente utile nella formazione dei menù.

```
DO CASE
CASE SCELTA=1
      DO ORDINI.CMD
CASE SCELTA=2
      DO BOLLE.CMD
CASE SCELTA=3
      DO FATTURE.CMD
CASE SCELTA=4
      DO TRATTE.CMD
CASE SCELTA=5
      DO MAG.CMD
CASE SCELTA=6
      ERASE
      SET CONSOLE OFF
      QUIT
ENDCASE
```

Anche in questo caso ad ogni comando DO CASE corrisponde un ENDCASE.

7.3.3 La esecuzione ripetuta: DO WHILE ... ENDDO.

La ripetizione di una serie di comandi, finchè è vera una data condizione, accertata a inizio ciclo, viene realizzata mediante i comandi:

```
DO WHILE CONDIZIONE
      COMANDI
ENDDO
```

Il tipo di ripetizione desiderata nel programma rende necessario l'impiego di tipi particolari di condizioni di ripetizione che possono essere qui di seguito elencati:

Tipo di ripetizione	impiego	condizione di ripetizione
Ciclo senza fine con uscita per abbandono di DBII	nel menu' del programma principale	DO WHILE T ..... COMANDI QUIT ENDDO
Ciclo con termine a fine file	nei programmi di lettura o correzione dei file record per record	DO WHILE .NOT. EOF COMANDI ENDDO

Ciclo con variabile di controllo	nei programmi a ripetizione controllata dallo utente e ritorno ad altro programma	STORE T TO ALTRO
		DO WHILE ALTRO
		COMANDI
		IF CONDIZIONE
		STORE F TO ALTRO
		ENDIF
		ENDDO

Ciclo a ripetizione enumerata	nei programmi a ripetizione controllata da contatore	STORE 10 TO CONT DO WHILE CONT >0 COMANDI STORE CONT-1 TO CONT ENDDO

Tabella D7.1

7.3.4 Il richiamo della esecuzione di un altro file.

Dall'interno di un file di comandi è possibile richiamarne altri con un massimo di 15, per cui è possibile avere 16 file aperti simultaneamente. Il controllo dai file chiamati viene restituito al file chiamante quando DBII incontra il comando RETURN oppure il segnale di fine file.

I file.CMD vengono chiamati con il comando:

DO <NOME FILE.CMD>

il file chiamato viene immediatamente messo in esecuzione.

I file di formati vengono aperti con il comando:

SET FORMAT TO <NOME FILE>

Il formato viene visualizzato all'atto della esecuzione del comando READ.

CAPITOLO 8

UN PROGRAMMA: STRUTTURA GENERALE

8.1 I programmi: struttura per funzioni.
8.2 Le funzioni generali.
8.3 Le funzioni specifiche.
8.4 La definizione delle fasi di realizzazione di un programma.

8.1 I programmi: struttura per funzioni

In una procedura è possibile individuare una serie di funzioni distinte che insieme composte consentono di raggiungere lo scopo complesso per cui il programma è stato progettato.

Si tratta di funzioni legate ad esigenze generali comuni a tutti i programmi, gestione dei file, gestione delle maschere video e di stampa, inizializzazione delle variabili di memoria, controlli di input e di funzioni specifiche caratteristiche di ciascun programma, connesse allo scopo complesso da raggiungere, l'aggiornamento degli articoli in stock, il calcolo della scorta minima, la emissione e la registrazione delle bolle e delle fatture, l'emissione delle tratte etc.

Si ritiene di poter affermare quanto segue:

1) ogni funzione, generale o specifica, deve essere realizzata mediante un distinto modulo di programma;

2) ogni funzione specifica può servirsi di più moduli generali;

3) le funzioni specifiche debbono essere collegate da un modulo generale di controllo a menù.

8.2 Le funzioni generali

Le funzioni generali sono funzioni strumentali rispetto agli scopi ultimi della procedura: la registrazione e la stampa di un giornale di movimenti (scopi della procedura) richiede la esecuzione di moduli di programma di gestione dei file, di gestione maschere, di controllo, strumentali rispetto ai primi.

La gestione dei file, la scelta e inizializzazione delle variabili di memoria, la creazione

della maschere possono essere affrontate solo dopo che è stata effettuata una analisi del problema da risolvere con la individuazione degli output, la definizione degli input, la definizione delle elaborazioni da fare.

Si prescinda per ora da un problema concreto e si immagini questa analisi già effettuata. Si ritiene che le funzioni generali di programma possano essere riassunte e analizzate nel modo seguente:

1) GESTIONE FILE DI DATI:
   creazione della struttura;
   aggiornamento;
   ricerca;
   visualizzazione e/o stampa;
   modifica;
   cancellazione;
   copia.

2) GESTIONE MASCHERE:
   creazione maschere video;
   creazione maschere di stampa.

3) INIZIALIZZAZIONE DELLE VARIABILI DI MEMORIA.

4) GESTIONE DEL CONTROLLO DEI SINGOLI MODULI DI PROGRAMMA.

8.3 Le funzioni specifiche

Le funzioni specifiche sono costituite dagli obiettivi ultimi per cui il programma è richiesto e possono essere definiti solo in rapporto a un programma concreto, in stretto rapporto con la individuazione degli "output richiesti".

Da questi dipende la possibilità di definire in concreto "input" ed "elaborazioni", quindi caratteristiche dei file di dati e dei file di variabili di memoria da utilizzare, caratteristiche delle maschere video e di stampa.

Si consideri ora una procedura concreta nei suoi obiettivi.

Si tratti di una procedura di magazzino; i suoi obiettivi, o le sue funzioni specifiche, possono essere considerati i seguenti:

- registrazione dei movimenti di magazzino;
- lista dei movimenti;
- revisione dei movimenti;
- stampa giornale dei movimenti;
- *aggiornamento articoli*;
- stampa articoli sottoscorta;
- stampa progressivi di magazzino;
- azzeramento e riapertura;
- valutazione scorte finali;
- stampa inventari.

8.4 La definizione delle fasi di realizzazione di un programma

Si immagini di dover preparare nel concreto i moduli di programma per la realizzazione del punto 5 "aggiornamento articoli".

Si definiscono in primo luogo le fasi di realizzazione del progetto nella seguente successione:

I CREAZIONE DELLA STRUTTURA DEI FILE DI DATI.

Si nota che in DBII la creazione dei file può essere realizzata in "immediato" in modo interattivo e che quindi questa operazione può essere effettuata a parte, prima di qualsiasi altra, con il comando CREATE.

II CREAZIONE DELLE MASCHERE VIDEO.

Si nota che sia il programma principale di controllo, che quelli specifici di gestione file abbisognano di maschere video.

III PREPARAZIONE DEL PROGRAMMA DI INIZIALIZZAZIONE DELLE VARIABILI

Si ricorda che la utilizzazione delle maschere richiede la previa inizializzazione delle variabili impiegate nelle maschere. Si richiede pertanto un programma di inizializzazione.

IV PREPARAZIONE DEL MODULO GENERALE DI CONTROLLO.

Il modulo generale di controllo è costituito dai comandi di richiamo dei programmi di inizializzazione, di maschera video e di una struttura DO CASE...ENDCASE di richiamo degli specifici programmi di gestione file.

V PREPARAZIONE DEI PROGRAMMI SPECIFICI

I programmi specifici potranno essere costituiti dai comandi di richiamo delle maschere video, con qualche opzione a menù, da utilizzare, eventualmente da un comando di richiamo della inizializzazione di variabili e da comandi di richiamo di altri sottoprogrammi, quindi dalla serie dei comandi di elaborazione dati e da un comando di ritorno al programma principale.

CAPITOLO 9

LA REALIZZAZIONE DEI SINGOLI MODULI DI PROGRAMMA

9.1 La creazione della struttura dati.
9.2 La codifica delle maschere video:
9.2.1 la codifica della maschera video generale;
9.2.2 la codifica delle maschere specifiche;
9.2.3 la creazione in "immediato" delle maschere di stampa.
9.3 La codifica del modulo di inizializzazione delle variabili.
9.4 La codifica del modulo generale di controllo.
9.5 La codifica dei moduli specifici.
9.6 La codifica dei moduli che utilizzano più file: la registrazione dei movimenti di magazzino.

9.1 La creazione della struttura dati

Dati gli scopi della nostra procedura si può ritenere che i file da essa direttamente interessati siano un file degli articoli a magazzino con tutte le caratteristiche quantitative e qualitative, un file di giacenze per periodo di formazione per la valutazione delle giacenze, un file di movimenti degli articoli per la registrazione dei movimenti e un file tabelle per le causali dei movimenti stessi e i codici IVA.

Si impiega in immediato il comando CREATE; i nomi attribuiti ai file sono rispettivamente MACOR, MASTOR, MOVAR e TABELLE; la struttura prescelta risulta dalle

(video 9.1).

```
. USE MACOR.DBF
. DISPLAY STRUCTURE
STRUCTURE FOR FILE:   MACOR.DBF
NUMBER OF RECORDS:    00006
DATE OF LAST UPDATE:  10/18/82
PRIMARY USE DATABASE
FLD        NAME        TYPE WIDTH    DEC
001        CODART       N    006
002        DESC         C    030
003        QTI          N    006
004        QTE          N    006
005        VCAPPR       N    011
006        COSTOU       N    008
007        QTU          N    006
008        SCORMTN      N    006
009        DATULV       C    008
010        DATULA       C    008
011        QTISCT       N    006
012        SCTI         N    004    002
013        QTIISCT      N    006
014        SCTII        N    004    002
015        PREZZOL      N    008
016        IVA          N    004    002
** TOTAL **             00128
.
```

```
. DISP STRU
STRUCTURE FOR FILE:   MASTOR.DBF
NUMBER OF RECORDS:    00000
DATE OF LAST UPDATE:  08/01/83
PRIMARY USE DATABASE
FLD        NAME        TYPE WIDTH    DEC
001        CODART       N    006
002        PERFOR1      C    008
003        INCPER1      N    006
004        VLUINC1      N    007
005        PERFOR2      C    008
006        INCPER2      N    006
007        VLUINC2      N    007
008        PERFOR3      C    008
009        INCPER3      N    006
010        VLUINC3      N    007
011        PERFOR4      C    008
012        INCPER4      N    006
013        VLUINC4      N    007
014        PERFOR5      C    008
015        INCPER5      N    006
016        VLUINC5      N    007
** TOTAL **             00112
```

```
. USE MOVART
. DISPLAY STRUCTURE
STRUCTURE FOR FILE:   MOVART.DBF
NUMBER OF RECORDS:    00006
DATE OF LAST UPDATE:  03/09/84
PRIMARY USE DATABASE
FLD        NAME        TYPE WIDTH    DEC
001        CODARMOV     N    006
002        QTEMOV       N    006
003        QTUMOV       N    006
004        PREZZUMOV    N    006
005        PREZCMOV     N    009
006        SCONTOMOV    N    009
007        IVA%         N    004    002
008        IVAMOV       N    009
009        NUMORMOV     N    005
010        NUMBOLMO     N    005
011        CODCAUS      N    004
012        DATMOV       N    008
** TOTAL **             00078
.
```

```
. USE TABELLE
. DISPLAY STRUCTURE
STRUCTURE FOR FILE:   TABELLE.DBF
NUMBER OF RECORDS:    00006
DATE OF LAST UPDATE:  10/18/82
PRIMARY USE DATABASE
FLD        NAME        TYPE WIDTH    DEC
001        CODMOV       N    002
002        CAUSALI      C    030
003        CODJV        N    003
004        TITES        C    030
005        AL           N    005    002
** TOTAL **             00071
.
```

Video 9.1

9.2 La codifica delle maschere video:

9.2.1 La codifica di una maschera video generale.

Si ritiene di poter utilizzare una maschera video generale così congegnata:

- un titolo di maschera in testa;
- un corpo di maschera per l'input output di ciascun programma utente;

- tre righe di video a disposizione per i menù
- una riga a disposizione per la risposta al menù.

Le variabili utilizzate potranno essere le seguenti:

TITOLO per il titolo della maschera;
PROMPT1, PROMPT2, PROMPT3 per i messaggi di menù;
SCELTA per le risposte dell'operatore.

La registrazione del programma relativo può essere effettuata sia ricorrendo ad una utility di DBII che mediante il comando MODIFY COMMAND <nome file. FMT>.

Con la utility è possibile creare la maschera scegliendo con un cursore le posizioni di scrittura , scrivendone la parte costante direttamente da tastiera su video, facendo procederei nomi dellevariabili di output da > e quelle di input da <.

Con il comando MODIFY COMMAND <nome file. FMT> si provvede a registrare i comandi seguenti:

```
********* SALVATO COME MENUGEN.FMT ***************
** Richiede la inizializzazione delle variabili di memoria
*** TITOLO,PROMPT1,PROMPT2,PROMPT3 E SCELTA.
@1,30 SAY TITOLO
@2,2 SAY "_____________________________________________________________"

@17,2 SAY"*****************************************************************"
@18,2 SAY "|"
@18,5 SAY PROMPT1
@18,78 SAY "|"
@19,2 SAY "|"
@19,5 SAY PROMPT2
@19,78 SAY "|"
@20,2 SAY "|"
@20,5 SAY PROMPT3
@20,78 SAY "|"
@21,2 SAY"|                                DAMMI LA TUA SCELTA"
@21,78 SAY "|"
@21,55 GET SCELTA
@21,78 SAY "|"
@22,2 SAY"*****************************************************************"
```

File 9.1

9.2.2 La codifica delle maschere video specifiche.

L'"aggiornamento articoli" per il quale lavoriamo richiede la realizzazione di tanti moduli distinti quante sono le funzioni di gestione file da realizzare e in particolare aggiornamento, ricerca, visualizzazione o stampa, modifica, cancellazione, copia. Ognuno di questi moduli richiede l'impiego di maschere video alcune delle quali possono essere utilizzate in comune. In particolare la visualizzazione richiederà una maschera per la visualizzazione dei dati, la modifica e l'aggiornamento richiederanno oltre a questa una maschera per l'inserimento. Complessivamente si richiedono le maschere seguenti:

1) maschere di menù - Si sfrutta la parte menù della
- maschera generale.

2) maschera di visualizzazione dati - Si sfrutta il menù della
- maschera generale, si aggiungono al suo corpo i comandi SAY

3) maschera di input dati - Si sostituiscono comandi GET
- a SAY della maschera di visualizzazione.

Per le maschere di menù, avendo già registrato la mascherà generale, non vi è nessuna operazione da fare, si presenta invece la codifica delle altre due maschere (file d 9.2, d 9.3) fatta tenendo conto della struttura dati del file MACOR. DBF. e le videate risultanti. Si nota che i nomi delle variabili utilizzate per le maschere corrisponde al nome delle variabili di file precedute da una lettera qualsiasi per comodità di memorizzazione.

```
*SALVATO COME VIDEOOUT.FMT
*Serve il programma di registrazione dati AGGIUNTA
*Richiede il richiamo delle variabili di memoria  da MACOR.MEM
@1,30 SAY TITOLO
@2,2 SAY"_____________________________________________________________"
********* PARTE SPECIFICA DELLA VIDEATA ****************
@4,5 SAY "CODICE ARTICOLO"
@4,20 SAY CODART
@5,5 SAY "QTA' INIZIALE"
@5,18 SAY QTI
@5,30 SAY"QTA' ENTRATA"
@5,42 SAY QTE
@5,54 SAY "QTA' USCITA"
@5,65 SAY QTU
@7,10 SAY "COSTO UNITARIO L."
@7,27 SAY COSTOU
@7,42 SAY "V.PROGRESSIVO DI CARICO"
@7,62 SAY VCARPR
@9,30 SAY"SCORTA MINIMA"
@9,44 SAY SCORMIN
@10,10 SAY"DATA ULTIMO ACQUISTO"
@10,30 SAY DATULA
@10,40 SAY "DATA ULTIMA VENDITA"
@10,60 SAY DATULV
@12,5 SAY"QTA' I SCONTO"
@12,19 SAY QTISCT
@12,32 SAY "SCONTO"
@12,40 SAY SCTII
@13,32 SAY "QTA' II SCONTO"
@13,45 SAY QTIISCT
@13,58 SAY "SCONTO"
@13,65 SAY SCTII
@14,20 SAY "PREZZO DI LISTINO"
@14,44 SAY PREZZOL
@15,20 SAY "IVA"
@15,26 SAY IVA
@16,10 SAY "DESCRIZIONE"
@16,25 SAY DESC
********* PARTE COMUNE A TUTTE LE VIDEATE **********
@17,2 SAY"*************************************************************"
```

```
@18,2 SAY "|"
@18,5 SAY PROMPT1
@18,78 SAY "|"
@19,2 SAY "|"
@19,5 SAY PROMPT2
@19,78 SAY "|"
@20,2 SAY "|"
@20,5 SAY PROMPT3
@20,78 SAY "|"
@21,2 SAY"|                                   DAMMI LA TUA SCELTA"
@21,78 SAY "|"
@21,58 GET SCELTA
@21,78 SAY "|"
@22,2 SAY"*************************************************************"
```

File 9.2

```
A:TYPE VIDEOIN.FMT
*SALVATO COME VIDEOIN.FMT
*Serve il programma di registrazione dati AGGIUNTA
*Richiede il richiamo delle variabili di memoria  da MACOR.MEM
@1,30 SAY TITOLO
@2,2 SAY "_____________________________________________________________________"
********* PARTE SPECIFICA DELLA VIDEATA ***************
@4,5 SAY "CODICE ARTICOLO"
@4,20 GET MCODART
@5,5 SAY "QTA' INIZIALE"
@5,18 GET MQTI
@5,30 SAY"QTA' ENTRATA"
@5,42 GET MQTE
@5,54 SAY "QTA' USCITA"
@5,65 GET MQTU
@7,10 SAY "COSTO UNITARIO L."
@7,27 GET MCOSTOU
@7,42 SAY "V.PROGRESSIVO DI CARICO"
@7,62 GET MVCARPR
@9,30 SAY"SCORTA MINIMA"
@9,44 GET MSCORMIN
@10,10 SAY"DATA ULTIMO ACQUISTO"
@10,30 GET MDATULA
@10,40 SAY "DATA ULTIMA VENDITA"
@10,60 GET MDATULV
@12,5 SAY"QTA' I SCONTO"
@12,19 GET MQTISCT
@12,32 SAY"SCONTO"
@12,40 GET MSCTI
@13,32 SAY"QTA' II SCONTO"
@13,45 GET MQTIISCT
@13,58 SAY "SCONTO"
@13,65 GET MSCTII
@14,20 SAY "PREZZO DI LISTINO "
@ 14,44 GET MPREZZOL
@ 15,20 SAY "IVA"
@15,24 GET MIVA
@16,10 SAY "DESCRIZIONE"
@16,25 GET MDESC
********* PARTE COMUNE A TUTTE LE VIDEATE **********
@17,2 SAY"*****************************************************************"
@18,2 SAY "|"
@18,5 SAY PROMPT1
@18,78 SAY "|"
@19,2 SAY "|"
@19,5 SAY PROMPT2
@19,78 SAY "|"
```

File 9.3 (segue)

```
@20,2 SAY "|"
@20,5 SAY PROMPT3
@20,78 SAY "|"
@21,2 SAY"|                                    DAMMI LA TUA SCELTA"
@21,78 SAY "|"
@21,58 GET SCELTA
@21,78 SAY "|"
@22,2 SAY"*****************************************************************"
```

File 9.3

```
                              AGGIUNGI
-----------------------------------------------------------------------------

   CODICE ARTICOLO:          0:
   QTA' INIZIALE:          0:QTA' ENTRATA:          0:QTA' USCITA:          0:

        COSTO UNITARIO L.:          0:   V.PROGRESSIVO DI CAR:          0:

                              SCORTA MINIMA :          0:
        DATA ULTIMO ACQUISTO:        :DATA ULTIMA VENDITA :         :

   QTA' I SCONTO :          0: SCONTO  :          0:
                                QTA' II SCONT:          0: SCONTO :          0:
                  PREZZO DI LISTINO        :          0:
                  IVA :          0:
        DESCRIZIONE    :                          :
*****************************************************************************
|       UN ALTRO ANCORA?                                                    |
|  S)i-N)o                                                                  |
|                                                                           |
|                              DAMMI LA TUA SCELTA     : :                  |
*****************************************************************************
```

Video 9.2

Il numero dei dati da visualizzare è piuttosto grande per cui è evidente che la videata non è esemplare. Sarebbe quindi opportuno spezzare in due parti sia la maschera di visualizzazione, che quella di input.

9.2.3 La creazione in "immediato" delle maschere di stampa.

La creazione di maschere di stampa può essere effettuata ricorrendo al comando in "immediato" REPORT FORM <nome file> che consente di stabilire interattivamente margini di stampa, linee per pagina, intestazione, variabili da stampare e posizioni di stampa. Si veda il seguente esempio (video 9.3).

```
A:DBASE

. USE MACOR
.
. REPORT FORM ARTICOLI.FRM
ENTER OPTIONS, M=LEFT MARGIN, L=LINES/PAGE, W=PAGE WIDTH L=55,W=74
PAGE HEADING? (Y/N) Y
ENTER PAGE HEADING: LISTA ARTICOLI DISPONIBILI
DOUBLE SPACE REPORT? (Y/N) N
ARE TOTALS REQUIRED? (Y/N) N
COL      WIDTH,CONTENTS
001      8,CODART
ENTER HEADING: COD.ART.
002      30,DESC
ENTER HEADING: DESCRIZIONE ARTICOLI
003      11,QTI+QTE-QTU
ENTER HEADING: DISPONIBILE
004      10,PREZZOL
ENTER HEADING: PREZZO
005      6
SYNTAX ERROR,  RE-ENTER
005      6,IVA
ENTER HEADING: IVA
```

Video 9.3

```
PAGE NO. 00001
10/18/82

                    LISTA ARTICOLI DISPONIBILI

COD.ART.      DESCRIZIONE ARTICOLI      DISPONIBILE   PREZZO     IVA

     101 CNOSSO BOX IN PVC DA 35*20*5          517      18500    0.18
     105 CRETA BOX IN PVC DA 40*27*10          377      19800    0.18
     107 DELFO BOX DA 50*30*28 IN PVC          150      21500    0.18
     109 ANTEA LUX DA 30*20*5 IN PVC           300      17000    0.18
     102 OLIMPIA MAXIBOX IN PLV DA 30*3        753      24500    0.18
     103 CLIZIA VIP IN VPC 30*10*20            133      35000    0.18
```

Video 9.4

9.3 La codifica del modulo di inizializzazione delle variabili

Le variabili da inizializzare sono quelle che dovranno essere impiegate per le maschere video quali variabili temporanee delle variabili di file dati e come variabili di calcolo. Si decide, per facilitarne la memorizzazione, di dare alle variabili di memoria gli stessi nomi, preceduti da una "m", dati alle variabili di file. La codifica è la seguente (file d 9.4) :

```
A:TYPE INIZIA.CMD
*E' SALVATO COME INIZIA.CMD
*IL PROGRAMMA E' CHIAMATO DAL PROGRAMMA MAIN.CMD
*HA LA FUNZIONE DI SALVARE LE VARIABILI DI MEMORIA
*PER L'USO DEI FILE DBF DI MAGAZZINO CORRENTE
  SET TALK OFF
  SET INTENSITY OFF
    IF .NOT. FILE( "MACOR.MEM" )
        STORE "                        " TO TITOLO
        STORE 0 TO MCODART
        STORE 0 TO MQTI
        STORE 0 TO MQTE
        STORE 0 TO MVCARPR
        STORE 0 TO MCOSTOU
        STORE 0 TO MQTU
        STORE 0 TO MSCORMIN
        STORE "         " TO MDATULV
        STORE "         " TO MDATULA
        STORE 0 TO MQTISCT
        STORE 0 TO MQTIISCT
        STORE 0.00 TO MSCTI
        STORE 0.00 TO MSCTII
        STORE 0 TO MPREZZOL
        STORE 0.00 TO MIVA
        STORE"                                                    "TO MDESC
        STORE"                                                    "TO SCELTA
             SAVE TO MACOR
             RELEASE ALL
    ENDIF
  USE MACOR INDEX DESC,CODART
```

File 9.4

9.4 La codifica del modulo generale di controllo

Il modulo generale di controllo provvede a richiamare il programma di salvataggio delle variabili di memoria, chiama il formato-video e assegna i dati del menù alle variabili di maschera. Presenta il menù delle opzioni possibili e mette in esecuzione il modulo richiestogli dall'operatore (file d 9.5).

```
A:TYPE AGGIORNA.CMD
*********************** SALVATO COME AGGIORNA.CMD ************************
***Richiama il programma di salvataggio delle variabili di memoria
***Presenta il menu' di aggiornamento, lo legge e mette in esecuzione
***la scelta dell'operatore
***Utilizza le variabili locali TITOLO, PROMPT1, PROMPT2, PROMPT3 e
***SCELTA.
**                                                                  **
*Elimina i messaggi di sistema di DBII
SET TALK OFF
DO INIZIA.CMD
DO WHILE T
   SET FORMAT TO MENUGEN
       STORE "AGGIORNAMENTO ARTICOLI" TO TITOLO
       STORE "A)ggiungi, R)icerca, V)isualizza o stampa" TO PROMPT1
       STORE "M)odifica, E)limina, C)opia" TO PROMPT2
```

```
        STORE "T)orna al menu'principale, F)ine lavoro" TO PROMPT3
        STORE "  " TO SCELTA
*  Si provvede alla lettura del menu' scelto
    READ
*  Si esegue la funzione scelta
    DO CASE
            CASE SCELTA="A"
            DO AGGIUNGI.CMD
            CASE SCELTA="R"
            DO RICERCA.CMD
            CASE SCELTA="V"
            DO GUARDA.CMD
            CASE SCELTA="M"
            DO MODIFICA.CMD
            CASE SCELTA="E"
            DO ELIMINA.CMD
            CASE SCELTA="C"
            DO COPIA.CMD
            CASE SCELTA="T"
            DO MAIN.CMD
            CASE SCELTA="F"
*    Pulisci lo schermo
            ERASE
*    Abbandona DBII
            QUIT
    ENDCASE
 ENDDO
```

File 9.5

9.5 La codifica dei moduli specifici: AGGIUNGI, RICERCA, MODIFICA, STAMPA, ELIMINA (file d 9.6, d 9.7, d 9.8, d 9.9, d 9.10)

```
A:TYPE AGGIUNGI.CMD
**** SALVATO COME AGGIUNGI.CMD
*Questo programma consente di aggiungere nuovi articoli di magazzino
*nell'archivio MACOR.DBF.
*Definisce il format-video da utilizzare
*Inizializza le variabili di memoria
*Se il primo o l'ultimo campo non sono nulli sostituisce
*Rimette nello stato iniziale la variabile SCELTA
*Restituisce il controllo al programma principale
**                                                          ***
  SET FORMAT TO VIDEOIN
  STORE "S" TO SCELTA
    DO WHILE SCELTA ="S"
*     Inizializza le variabili di memoria  *
        RESTORE FROM MACOR.MEM
        STORE "AGGIUNGI" TO TITOLO
        STORE"Un altro ?" TO PROMPT1
        STORE "                    S)i o N)o" TO PROMPT2
        STORE" "TO PROMPT3
        STORE" "TO SCELTA
        READ
          IF MCODART <> 0.OR. MDESC <>" "
             APPEND BLANK
               REPLACE CODART WITH MCODART, QTI WITH MQTI, QTE WITH MQTE
               REPLACE VCARPR WITH MVCARPR, COSTOU WITH MCOSTOU
               REPLACE QTU WITH MQTU, SCORMIN WITH MSCORMIN
               REPLACE DATULV WITH MDATULV, DATULA WITH MDATULA
```

File 9.6

```
            REPLACE QTISCT WITH MQTISCT,SCTI WITH MSCTI
            REPLACE QTIISCT WITH MQTIISCT,SCTII WITH MSCTII
            REPLACE PREZZOL WITH MPREZZOL, IVA WITH MIVA
            REPLACE DESC WITH MDESC
         ENDIF
   ENDDO
 INDEX ON DESC TO DESC
 INDEX ON CODART TO CODART
 STORE "A" TO SCELTA
RETURN
```

```
A:TYPE RICERCA.CMD
*************************** SALVATO COME RICERCA.CMD ***************************
** La ricerca viene effettuata sulla base dei primi caratteri di descri=
** zione di ogni articolo nel file di dati MACOR.DBF e nel file di indici
** relativo sulla chiave DESC.
**                                                                          **
   SET FORMAT TO MENUGEN
     STORE "RICERCA" TO TITOLO
     STORE "Dammi la CHIAVE per cercare o STOP per uscire" TO PROMPT3
     STORE " " TO PROMPT2
     STORE " " TO PROMPT1
     STORE "                                                    " TO SCELTA
   READ
   DO WHILE SCELTA # "STOP"
   STORE !(SCELTA) TO SCELTA
   STORE TRIM(SCELTA) TO SCELTA
** Effettua la ricerca sulla base del valore della variabile***
   FIND &SCELTA
** Se il valore del puntatore al record # e' zero, la chiave non e' valida
         IF # = 0
            STORE "NON TROVATO" TO PROMPT2
         ELSE
            SET FORMAT TO VIDEOOUT
            STORE " " TO PROMPT2
         ENDIF
      STORE"                                              " TO SCELTA
    READ
 ENDDO
** Ritorna al file chiamante.*****
RETURN
```

File 9.7

```
A:TYPE MODIFICA.CMD
***** SALVATO COME MODIFICA.CMD ***************
*Questo programma, dopo che e' stato visualizzato un dato record,
*provvede alla sua modifica.
*Utilizza le variabili locali gia' viste in VIDEOIN.FMT e le salva
*in un file di memoria di deposito temporaneo,quindi modifica il record
*con i dati forniti dall'operatore.
*                                                                        **
* Salva le variabili di lettura del file di dati
SAVE TO TEMP
*Assegna alle variabili di memoria quelle di lettura file
  STORE CODART TO MCODART
  STORE QTI TO MQTI
  STORE QTE TO MQTE
  STORE VCARPR TO MVCARPR
  STORE COSTOU TO MCOSTOU
  STORE QTU TO MQTU
  STORE SCORMIN TO MSCORMIN
  STORE DATULV TO MDATULV
```

File 9.8

```
  STORE DATULA TO MDATULA
  STORE QTISCT TO MQTISCT
  STORE SCTI TO MSCTI
  STORE QTIISCT TO MQTIISCT
  STORE SCTII TO MSCTII
  STORE PREZZOL TO MPREZZOL
  STORE IVA TO MIVA
  STORE DESC TO MDESC
  STORE "MODIFICA" TO TITOLO
  STORE " "TO PROMPT1, PROMPT2,PROMPT3, SCELTA
* Chiede conferma o modifica dei dati con VIDEOIN
    SET FORMAT TO VIDEOIN
    READ
* Sostituisce alle variabili di file quelle di memoria e registra
        REPLACE CODART WITH MCODART, QTI WITH MQTI, QTE WITH MQTE
        REPLACE VCARPR WITH MVCARPR, COSTOU WITH MCOSTOU
        REPLACE QTU WITH MQTU, SCORMIN WITH MSCORMIN
        REPLACE DATULV WITH MDATULV, DATULA WITH MDATULA
        REPLACE QTISCT WITH MQTISCT, SCTI WITH MSCTI
        REPLACE QTIISCT WITH MQTIISCT, SCTII WITH MSCTII
        REPLACE PREZZOL WITH MPREZZOL, IVA WITH MIVA
        REPLACE DESC WITH MDESC
* Rchiama le variabili di deposito temporaneo
  RESTORE FROM TEMP
RETURN
```

```
A:TYPE STAMPA.CMD

********************* SALVATO COME STAMPA.CMD ************************
** Serve a stampare record con numerosi campi su piu' di una riga per
** record.
**                                                                  **
SET TALK OFF
SET PRINT ON
GOTO TOP
DO WHILE .NOT. EOF
  * VA A INIZIO PAGINA
  EJECT
  ?
  ? "                LISTA DEGLI ARTICOLI IN MAGAZZINO"
  ? "*****************************************************************************'
  ?
  STORE 0 TO CONTAREC
  DO WHILE CONTAREC<10 .AND. (.NOT. EOF)
    ? "CODART", CODART ,"  DESCRIZIONE ",DESC
    ? "     QUANTITA' INIZIALE ",QTI
    ? "QUANTITA' ENTRATA =", "******* QUANTITA' USCITA'=",QTU
    ? "COSTO U.",COSTOU, "**PREZZO DI LISTINO=",PREZZOL,"** IVA=",IVA
    ? "SCORTA MINIMA",SCORMIN,"DATA ULTIMO AC.",DATULA,"DATA ULTIMA V.",DATULV
    ? "I SCONTO",SCTI,"QUANTITA'",QTISCT,"*II SCONTO",SCTII,"QUANTITA'",QTIISCT
    ? "                    VALORE PROGRESSIVO DI CARICO",VCARPR
    ? "-------------------------------------------------------------------"
    STORE CONTAREC+1 TO CONTAREC
    SKIP
   ENDDO
 SET PRINT OFF
 RELEASE CONTAREC
RETURN
```

File 9.9

```
A:TYPE ELIMINA.CMD
************** SALVATO COME ELIMINA.CMD ***********************************
*** Questo modulo serve per la cancellazione dei recordi di dati
SET TALK OFF
STORE "ELIMINA "TO TITOLO
STORE " " TO PROMPT1,PROMPT2
STORE "cerca A)vanti, cerca I)ndietro, M)arca per eliminare" TO PROMPT2
STORE "E)LIMINA ,F)ine lavoro" TO PROMPT3
STORE "                " TO SCELTA
SET FORMAT TO VIDEOOUT
READ
     DO WHILE SCELTA # "F"
        DELETE
          DO CASE
              CASE SCELTA="A"
                SKIP
              CASE SCELTA="I"
                SKIP-1
              CASE SCELTA="M"
                DELETE
              CASE SCELTA="E"
                STORE "ATTENZIONE ! ELIMINAZIONE DEFINITIVA" TO TITOLO
                DO WHILE .NOT. EOF
                   IF *
                      READ
                       IF SCELTA # "E"
                         RECALL
                       ENDIF
                      SKIP
                   ENDIF
                ENDDO
                PACK
          ENDCASE
        READ
     ENDDO
RETURN
```

File 9.10

9.6 La codifica di moduli che utilizzano più file: la REGISTRAZIONE DEI MOVIMENTI DI MAGAZZINO

Spesso è necessario utilizzare simultaneamente più file, ad esempio nella gestione di un magazzino per la registrazione dei suoi movimenti si deve utilizzare il "profile" o un file "tabelle" di causali di movimenti, il file dei movimenti da noi chiamato MOVART. DBF e il file degli articoli in magazzino MACOR. DBF.

Le causali dei movimenti dovranno essere utilizzate per tutto il periodo di durata della esecuzione del programma di registrazione. E' comodo mantenerne il contenuto in variabili di memoria per cui l'inizio del programma richiederà la lettura delle causali del file tabelle e la loro registrazione in un file di memoria, quindi richiederà la lettura del file di memoria. La codifica del programma viene qui di seguito presentata (file d 9.11):

```
A:TYPE INITMOV.CMD

*********************** SALVATO COM INITMOV.CMD ******************
***** E' CHIAMATO DA REGMOV. HA LA FUNZIONE DI SALVARE LE VARIABILI
***** DI MEMORIA PER LA REGISTRAZIONE DEI MOVIMENTI DI MAGAZZINO
***                                                              ***
SET TALK OFF
    IF .NOT. FILE ("MOV.MEM")
       STORE 0 TO MCODARMOV, MCODCAUS, MQTEMOV, MQTIMOV
       STORE 0 TO MDATMOV, MNUMORMOV, MNBOLMO, MIVAMOV
       STORE 0 TO  MIVA, MSCONTOMOV, MPREZZUMOV,MPREZCMOV
       USE TABELLE
           GOTO TOP
           STORE CAUSALI TO CAUSALE1
           SKIP
           STORE CAUSALI TO CAUSALE2
           SKIP
           STORE CAUSALI TO CAUSALE3
           SKIP
           STORE CAUSALI TO CAUSALE4
           SKIP
           STORE CAUSALI TO CAUSALE5
           SKIP
           STORE CAUSALI TO CAUSALE6
       SAVE TO MOV
       RELEASE ALL
  ENDIF
```

File 9.11

Se le variabili di memoria, per un massimo di 64, convivono con le variabili di file, non altrettanto succede per le variabili di un file in uso con quelle del file eventualmente usato precedentemente. Difatti il comando USE <nome file> provvede ad aprire il nuovo file e a chiudere il precedente di cui, in tal modo, perde le variabili e rende possibile l'impiego di più file successivamente, non l'impiego di due simultaneamente.

Per rendere possibile l'impiego di due file simultaneamente si deve dare a DBII il comando SELECT [PRIMARY] o SELECT [SECONDARY].

DBII mette a disposizione dei file due distinte aree di memoria: la prima è attiva

quando si mette in uso il primo file di dati da impiegare, la seconda viene attivata con i comandi SELECT [SECONDARY] e USE <nome file> e resterà tale fino a che non si richiama la prima. Per tale comando sarà possibile richiamare variabili di campo dall'una e dall'altra area rendendo possibile l'impiego simultaneo di due file.

Si nota che dopo i primi comandi di SELECT e USE la prima o la seconda area possono essere richiamate con il comando SELECT [ ] non seguito da USE.

Si presenta qui di seguito il modulo per la registrazione dei movimenti che impiega questa tecnica. Si presentano per completezza di funzionamento anche gli altri moduli di corredo (file d 9.12, d 9.13, d 9.14) con le relative videate (video d 9.4, d 9.5, d 9.6,d 9.7).

```
                                REGISTRAZIONE DELLE ENTRATE DI MAGAZZINO
-------------------------------------------------------------------------
```

```
*************************************************************************
|
|  DAMMI IL CODICE DELL'ARTICOLO
|
|                                  DAMMI LA TUA SCELTA  :102              :
*************************************************************************
```

```
********************** SALVATO COME REGEN.PRG ***************************
***** PROVVEDE ALLA REGISTRAZIONE DELLE ENTRATE                        ***
***** LEGGE IL CODICE DELL'ARTICOLO E LO RICERCA                       ***
***** NE VISUALIZZA I DATI                                             ***
***** PRENDE I DATI DI INPUT DA TASTIERA EFFETTUA I CONTEGGI           ***
***** REGISTRA OGNI NUOVO MOVIMENTO IN MOVART.DBF                      ***
***** AGGIORNA IL FILE SECONDARIO MACOR.DBF                            ***
***** E" CHIAMATO DA REGMOV.PRG                                        ***
***                                                                    ***
USE MOVART
SELECT SECONDARY
USE MACOR INDEX CODART
DO WHILE T
    SET FORMAT TO MENUGEN
        STORE "REGISTRAZIONE DELLE ENTRATE DI MAGAZZINO" TO TITOLO
        STORE "     " TO PROMPT1
        STORE "DAMMI IL CODICE DELL'ARTICOLO " TO PROMPT2
        STORE "  " TO PROMPT3
        STORE "                  " TO SCELTA
    READ
        STORE TRIM(SCELTA) TO SCELTA
        FIND &SCELTA
        IF # =0
           STORE" VUOI AGGIUNGERE QUESTO ARTICOLO  (S) O (N)" TO PROMPT2
           READ
```

```
        IF SCELTA = "S"
           DO AGGIUNGI.CMD
         ELSE
           DO REGMOV
        ENDIF
   ENDIF
   SET FORMAT TO MASENT
        DO WHILE SCELTA >"0"
           RESTORE FROM MOV
           STORE "REGISTRAZIONE ENTRATE DI MAGAZZINO" TO TITOLO
           STORE "     " TO PROMPT1
           STORE" DAMMI IL CODICE DELL'ARTICOLO" TO PROMPT2
           STORE" 0 PER USCIRE" TO PROMPT3
           STORE"                    "TO SCELTA
           STORE QTI+QTE-QTU TO ES
           READ
           SELECT PRIMARY
               STORE CODART TO MCODARMOV
             APPEND BLANK
               REPLACE CODARMOV WITH MCODARMOV, QTEMOV WITH MQTEMOV
               REPLACE PREZZUMOV WITH MPREZZUMOV, PREZCMOV WITH MPREZCMOV
               REPLACE NUMORMOV WITH MNUMORMOV,NUMBOLMO WITH MNBOLMO
               REPLACE CODCAUS WITH MCODCAUS, DATMOV WITH MDATMOV
           SELECT SECONDARY
              STORE QTE+MQTEMOV TO MQTEMOV
              STORE VCARPR+MPREZCMOV TO MPREZCMOV
              REPLACE QTE WITH MQTEMOV, VCARPR WITH MPREZCMOV
           STORE TRIM(SCELTA) TO SCELTA
           FIND &SCELTA
               IF # = 0
                 SET FORMAT TO MENUGEN
                  STORE " L'ARTICOLO NON ESISTE" TO PROMPT3
                  STORE " VUOI AGGIUNGERLO ?. (S) O (N)" TO PROMPT2
                 READ
                       IF SCELTA="S"
                          DO AGGIUNGI.CMD
                        ELSE
                          DO REGMOV
                       ENDIF
               ENDIF
           ERASE
        ENDDO
RETURN
```

File 9.12

```
                              REGISTRAZIONE ENTRATE DI MAGAZZINO
-----------------------------------------------------------------------
   N.BOLLA    :        105:        N.ORDINE   :        130:


   COD. ART. :        102
   DESCRIZIONE:   OLIMPIA MAXIBOX IN PLV DA 30*3
   GIACENZA          623
                                        COD.MOV      :          1:
   DATA       :    230384:              QUANTITA'    :        130:
    CODICI MOVIMENTI                    PREZZO U.    :      14500:
                                        PREZZO C.    :    1885000:
   1 ENTRATA-ACQUISTO DA FORN.
   3 ENTRATA - RESO DA CLIENTE
   5 ENTRATA - DA LAVORAZIONE

*********************************************************************
|
|   DAMMI IL CODICE DELL'ARTICOLO
|   0 PER USCIRE
|              DAMMI LA TUA SCELTA                        :                :
*********************************************************************
```

Video 9.5

```
A:TYPE MASENT.FMT
*SALVATO COME MASENT.FMT
1,30L
@2,2 SAY "_____________________________________________________________"
********* PARTE SPECIFICA DELLA VIDEATA ****************
@3,5 SAY "N.BOLLA"
@3,15 GET MNBOLMO
@3,35 SAY "N.ORDINE"
@3,45 GET MNUMORMOV
@6,5 SAY "COD. ART. :"
@6,20 SAY CODART
@7,5  SAY "DESCRIZIONE:"
@7,20 SAY DESC
@8,5  SAY "GIACENZA "
@8,15 SAY ES
@9,40 SAY "COD.MOV"
@9,52 GET MCODCAUS
@10,5 SAY "DATA"
@10,15 GET MDATMOV
@10,40 SAY "QUANTITA'"
@10,52 GET MQTEMOV
@11,5 SAY " CODICI MOVIMENTI"
@11,40 SAY "PREZZO U."
@11,52 GET MPREZZUMOV
@12,40 SAY "PREZZO C."
@12,52 GET MPREZCMOV
@13,5 SAY CAUSALE1
@14,5 SAY CAUSALE3
@15,5 SAY CAUSALE5
********* PARTE COMUNE A TUTTE LE VIDEATE **********
@17,2 SAY"*****************************************************************"
@18,2 SAY "|"
@18,5 SAY PROMPT1
@18,78 SAY "|"
@19,2 SAY "|"
@19,5 SAY PROMPT2
@19,78 SAY "|"
@20,2 SAY "|"
@20,5 SAY PROMPT3
@20,78 SAY "|"
@21,2 SAY"|"
@21,2 SAY "|              DAMMI LA TUA SCELTA"
@21,78 SAY "|"
@21,58 GET SCELTA
@21,78 SAY "|"
@22,2 SAY"*****************************************************************"
```

File 9.13

```
A:TYPE REGMOV.PRG
********************** SALVATO COME REGMOV.PRG ****************************
**** E' IL MODULO DI CONTROLLO DELLE REGISTRAZIONI DEI MOVIMENTI ********
****                                                                  ****
DO INITMOV.CMD
SET TALK OFF
   SET FORMAT TO MENUGEN
       STORE "REGISTRAZIONE DEI MOVIMENTI DI MAGAZZINO"TO TITOLO
       STORE "   " TO PROMPT1, PROMPT2
       STORE " E)ntrate, U)scite ,F)ine lavoro ?" to PROMPT3
       STORE "    " TO SCELTA
   READ
   DO CASE
      CASE SCELTA="E"
           DO REGEN
```

```
      CASE SCELTA="U"
           DO REGUS.CMD
      CASE SCELTA="F"
           QUIT
   ENDCASE
RETURN
```

File 9.14

```
                              MODIFICA
-------------------------------------------------------------------------------

   CODICE ARTICOLO:      10098:
   QTA' INIZIALE:        200:QTA' ENTRATA:          0:QTA' USCITA:          0:

        COSTO UNITARIO L.:      50000:   V.PROGRESSIVO DI CAR:  10000000:

                              SCORTA MINIMA :         50:
        DATA ULTIMO ACQUISTO:201283  :DATA ULTIMA VENDITA :         :

   QTA' I SCONTO :         10: SCONTO  :       0.00:
                              QTA' II SCONT:         20: SCONTO :       0.20:
                  PREZZO DI LISTINO        :      70000:
                  IVA :       0.18:
        DESCRIZIONE    :FLEX DD SCATOLE DA 10*5POL     :
*******************************************************************************
|                                                                             |
|                                                                             |
|                                                                             |
|                              DAMMI LA TUA SCELTA      : :                   |
*******************************************************************************
```

Video 9.6

```
                              REGISTRAZIONE DEI MOVIMENTI DI MAGAZZINO
     ------------------------------------------------------------------

     ******************************************************************
     |
     |
     |   E)ntrate, U)scite ,F)ine lavoro ?
     |                              DAMMI LA TUA SCELTA  :    :
     ******************************************************************
```

Video 9.7

PARTE IV

PROGRAMMI DI CALCOLO E SIMULAZIONE

Questa parte è stata realizzata in collaborazione da Patrizia Tomaselli Villa e Tullio Trincardi.

CAPITOLO 1

IL MULTIPLAN

1.1 I fogli di lavoro elettronici.
1.2 Il MULTIPLAN

1.1 I fogli di lavoro elettronici

Gli strumenti di rilevazione tradizionalmente impiegati nelle aziende forniscono valori prevalentemente consuntivi, ma soprattutto sono strutturati in modo da non prestarsi, se non dopo complesse elaborazioni, a divenire strumenti di decisione efficaci e rapidi. Questa difficoltà ha spesso costretto le direzioni aziendali a servirsi degli strumenti contabili a disposizione solo quali mezzi di rilevazione dei risultati raggiunti. Una tale destinazione limita fortemente la loro potenzialità informativa e li rende carenti ai fini decisionali.

Il calcolo per la soluzione in tempo reale dei problemi più disparati anche complessi, la simulazione di situazioni create da ipotetiche decisioni, in anticipo rispetto al momento in cui le decisioni reali si debbono prendere, costituiscono degli obiettivi importanti, problemi che non possono trovare una adeguata soluzione in programmi predisposti con i normali linguaggi di programmazione.

La importanza di questi obiettivi e la inadeguatezza delle soluzioni fino a pochi anni fa proposte ha indotto le aziende di software a produrre dei package di facilitazione di calcolo e di simulazione, alcuni dei quali hanno avuto grande successo (si pensi a VISICALC, SUPERCALC, MULTIPLAN, DOSSIER), modificando la qualità delle informazioni, incidendo in modo significativo nel modo di formazione delle decisioni.

Per rispondere alle esigenze di cui si è parlato tali programmi presentano alcune caratteristiche che li rendono versatili ed efficaci. Innanzitutto permettono di effettuare ripetutamente calcoli basati sui valori inseriti da tastiera e ne consentono il riepilogo tramite medie o totali. Inoltre consentono di memorizzare formule di calcolo; infine, per evidenziare il significato e le correlazioni esistenti tra i dati, permettono che essi vengano presentati in modo chiaro, semplice e facilmente interpretabile consentendo di visualizzare scritte illustrative dei dati inseriti.

La struttura che meglio di ogni altra risponde a queste esigenze è quella a *matrice*; in tale struttura, infatti, anche in assenza di titoli esplicativi la disposizione di valori costituisce già un'informazione, evidenzia già un legame tra i valori stessi.

La possibilità di memorizzare le formule e le relazioni tra i valori inseriti sui fogli elettronici e la relativa semplicità di utilizzo facilita la diffusione e l'impiego dei fogli elettronici nei più svariati campi di applicazione.

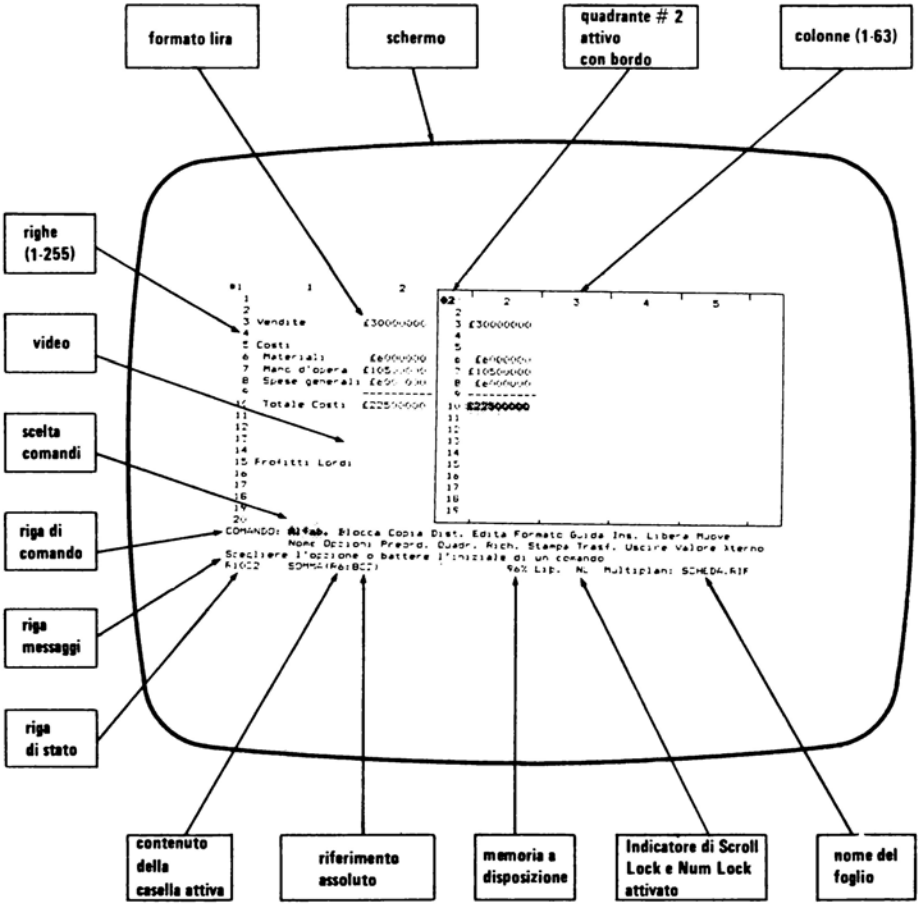
Questi fogli di lavoro si presentano come una larga matrice di righe e colonne che possono contenere parole, numeri o formule e che compongono insieme le funzioni dei prospetti manuali e delle calcolatrici, con la capacità di memorizzare i legami logico-

matematici tra gli input e di ricalcolare automaticamente i loro valori se qualcuno di essi viene variato. Attraverso un'esigua serie di comandi un foglio elettronico consente di risolvere in tempi brevissimi relazioni e legami tra valori anche notevolmente complessi.

1.2 Il MULTIPLAN

La matrice del MULTIPLAN IBM e TEXAS è composta di 255 righe e 63 colonne.

Una volta lanciato il MULTIPLAN appare sul monitor una porzione limitata (detta finestra) del foglio di lavoro, numerata per righe e per colonne. In particolare si potranno vedere 7 colonne e 20 righe (m 1.1).



M/1.1

Esistono anche Multiplan in italiano come quelli del Macintosh e dell'IBM il cui primo schermo viene qui sopra riportato.

Il cursore è posizionato nella casella della riga 1 colonna 1; si deve notare che la posizione del cursore viene evidenziata anche nella parte inferiore del monitor a sinistra con i caratteri R1C1. In linea con tale segnalazione si trova, procedendo a leggere verso destra, un numero che indica, percentualmente, lo spazio di memoria disponibile ed, infine, il nome del file che si sta utilizzando.

Nella parte inferiore appare tutta la serie dei comandi che è possibile attivare. Il *cursore di edit* è posizionato sul primo comando; per effettuare una opzione diversa è necessario spostare il cursore di edit sul comando desiderato mediante la barra spaziatrice e quindi dare il *return*, o selezionare l'iniziale del comando voluto e quindi il return.

Con il MULTIPLAN è possibile richiamare sul video altre finestre del foglio di lavoro al fine di visualizzarne il contenuto ed eventualmente intervenire su di esso.

Il cursore che si trova inizialmente posizionato in R1C1, può essere spostato utilizzando i tasti di direzione sia in senso verticale che orizzontale mantenendo premuto il tasto che muove il cursore verso destra, appariranno successivamente sullo schermo tutte le colonne fino alla sessantatresima casella. In modo analogo si può procedere per giungere fino alla duecentocinquantacinquesima riga.

Tuttavia, per rendere più veloce lo spostamento del cursore si può attivare il comando *GOTO*. Selezionato questo comando, compaiono, nella parte inferiore dello schermo, dei sub-comandi attraverso i quali si può indirizzare il cursore nella casella desiderata.

CAPITOLO 2

I COMANDI DEL MULTIPLAN

2.1 I comandi: classificazione.
2.2 Comandi di INPUT-OUTPUT
2.3 Comandi di formattazione dello schermo
2.4 Comandi di assegnazione e calcolo.

2. 1 I comandi: classificazione

I comandi di MULTIPLAN svolgono essenzialmente tre funzioni: funzioni di INPUT-OUTPUT, funzioni di formattazione dello schermo, funzioni di assegnazione e calcolo; naturalmente una classificazione di questo tipo non riesce a comprendere tutti i comandi che si possono attivare con il MULTIPLAN .

2.2 Comandi di INPUT-OUTPUT

I comandi fondamentali attraverso cui inserire parole, numeri, o formule sono il comando *ALPHA* e il comando *VALUE*. Optando per il primo appare nella parte inferiore dello schermo la scritta ALPHA seguita da due punti, ciò significa che il programma è pronto a ricevere il testo che si intende inserire nella casella in cui si trova il cursore.Dopo aver digitato il testo si preme return ed esso comparirà nelle casella voluta:

```
|                                                                |
|  ALPHA :                                                       |
|                                                                |
|  Enter text ( no double quotes )                               |
|  R1C1                        99% Free      Multiplan: TEMP     |
|                                                                |
-----------------------------------------------------------------
```

Con il comando VALUE ,invece, il MULTIPLAN si predispone ad accogliere un valore numerico o una formula.

Salvare in memoria la parte di foglio elettronico che si è costruita consente di interrompere il lavoro per poterlo poi riprendere esattamente nello stesso punto in cui si era abbandonato. Per effettuare questa operazione si usa il comando *TRANSFERT*. La linea di comando si presenta nel seguente modo:

```
|                                                                  |
|  TRANSFERT : Load Save Clear Delete Options Rename               |
|                                                                  |
|  Select option or type command letter                            |
|  R1C1                          99% Free     Multiplan: TEMP     |
-------------------------------------------------------------------
```

Per salvare il lavoro si seleziona S per SAVE:

```
|                                                                  |
|  TRANSFERT SAVE filename :TEMP                                   |
|                                                                  |
|  Enter a filename                                                |
|  R1C1                          99% Free      Multiplan: TEMP     |
-------------------------------------------------------------------
```

A questo punto occorre dare un nome abbastanza significativo al lavoro che si intende salvare in modo da facilitarne il richiamo in futuro. Operata questa scelta si preme "return" e il file viene salvato su disco.

Per disporre di una copia su carta del foglio di lavoro elettronico occorre azionare il comando *PRINT*:

```
|                                                                  |
|  PRINT :Printer File Margins Options                             |
|                                                                  |
|  Select option or type command letter                            |
|  R1C1                          99% Free      Multiplan: TEMP     |
-------------------------------------------------------------------
```

Il Multiplan ha fissati degli standard per i margini che possono però essere modificati secondo le necessità di stampa dell' utente. Queste variazioni si possono ottenere selezionando il sub-comando Margins:

```
|                                                                  |
|  PRINT MARGINS:left:5 top:6 print width:70  print length:54      |
|                page length:66                                    |
|  Enter a number                                                  |
|  R1C1                          99% Free     Multiplan: TEMP     |
-------------------------------------------------------------------
```

2.3 Comandi di formattazione dello schermo

Con il comando *FORMAT* è possibile modificare il formato di visualizzazione delle celle o del loro contenuto.

```
|                                                                  |
|  FORMAT :Cells Default Options Width                             |
|  Select option or type command letter                            |
|                                                                  |
|  R1C1                            99% Free  Multiplan: TEMP       |
-------------------------------------------------------------------
```

Supponiamo di voler modificare il formato di tutte le celle allora occorre selezionare Default e sulla riga dei comandi apparirà (m 2.7):

```
|                                                                  |
|  FORMAT DEFAULT :Cells Width                                     |
|  Select option or type command letter                            |
|                                                                  |
|  R1C1                             99% Free  Multiplan: TEMP      |
-------------------------------------------------------------------
```

Optando per Cells compare:

```
|                                                                  |
|  FORMAT DEFAULT CELLS alignment:Ctr Gen Left Right               |
|   format code : Cont Exp Fix(Gen)Int $ * % # of decimals 0       |
|  Select option                                                   |
|                                                                  |
|  R1C1                              99% Free  Multiplan: TEMP     |
-------------------------------------------------------------------
```

Nella prima riga è possibile scegliere l'allineamento desiderato, nella seconda il formato che si intende visualizzare.

Un comando di formattazione utile in quanto consente di bloccare descrizioni o formule è il comando *LOCK*:

```
|                                                                  |
|  LOCK : Cells Formulas                                           |
|  Select option or type command letter                            |
|                                                                  |
|  R1C1                           99% Free     Multiplan: TEMP     |
-------------------------------------------------------------------
```

Per proteggere ad esempio le formule che si sono inserite in un lavoro occorre selezionare l'opzione F per Formulas:

```
|                                                                |
|  LOCK FORMULAS :                                               |
|  Enter Y to confirm                                            |
|                                                                |
|  R1C1                          99% Free      Multiplan: TEMP   |
-----------------------------------------------------------------
```

Premendo Y e confermando l'opzione vengono bloccate tutte le formule contenute nelle celle.

Qualora interessi invece fissare delle etichette o titoli che restino visibili anche quando si fanno scorrere le colonne si usa il comando WINDOW:

```
|                                                                |
|  WINDOW : Split Border Close Link                              |
|  Select option or type command letter                          |
|                                                                |
|  R1C1                          99% Free      Multiplan: TEMP   |
-----------------------------------------------------------------
```

Per separare i titoli di una determinata finestra di lavoro dai dati in essa contenuti per mantenere fissi i primi si seleziona SPLIT:

```
|                                                                |
|  WINDOW SPLIT : Horizontal Vertical Titles                     |
|  Select option or type command letter                          |
|                                                                |
|  R1C1                          99% Free      Multiplan: TEMP   |
-----------------------------------------------------------------
```

Tra i diversi subcomandi che compaiono scegliamo Titles in quanto si intende fissare i titoli della colonna e della riga 2:

```
|                                                                |
|  WINDOW SPLIT TITLES :#of rows:2    #of columns:1              |
|  Enter a number                                                |
|                                                                |
|  R1C1                          99% Free      Multiplan: TEMP   |
-----------------------------------------------------------------
```

2.4 Comandi di assegnazione e calcolo

Il comando *COPY* consente di fissare le celle in cui si vuole copiare il contenuto della cella attiva. Selezionando la lettera C per COPY sulla parte bassa dello schermo apparirà:

```
|                                                                |
|   COPY : Right Down From                                       |
|   Select option or type command letter                         |
|                                                                |
|   R1C1                          99% Free      Multiplan: TEMP   |
-----------------------------------------------------------------
```

Qualora si opzioni R per Right comparirà:

```
|                                                                    |
|   COPY RIGHT number of cells:      starting at R1C1                |
|   Enter a number                                                   |
|                                                                    |
|   R1C1                            99% Free     Multiplan: TEMP     |
----------------------------------------------------------------------
```

A questo punto è sufficiente inserire il numero di celle desiderato e premere return per copiare il contenuto della cella R1C1.

Costruito un foglio di lavoro che contiene diverse formule ogniqualvolta viene modificato il contenuto di una cella il Multiplan ricalcola il valore di tutte quelle celle che dipendono da essa. Se il foglio di lavoro contiene molte formule, ogni cambiamento richiede un pò di tempo per completare il ricalcolo. Questa procedura si può velocizzare bloccando il ricalcolo automatico con il comando *OPTION*:

```
|                                                                    |
|  OPTIONS recalc : Yes No      mute : Yes No                        |
|          iteration : Yes No   completion test at :                 |
|  Select option                                                     |
|                                                                    |
|  R1C1                        99% Free         Multiplan: TEMP      |
----------------------------------------------------------------------
```

Si seleziona NO e viene bloccato il meccanismo di ricalcolo, quindi si preme "return".

CAPITOLO 3

LE FORMULE

3.1 Le formule
3.2 I modi di riferimento ai valori delle celle: il riferimento in modo assoluto e in modo relativo.
3.3 Gli operatori aritmetici.
3.4 Gli operatori logici
3.5 I principali operatori funzionali.

3.1 Le formule

In MULTIPLAN è possibile registrare formule in celle destinate ad accoglierne i risultati.

Queste formule operano su valori costanti, celle o gruppi di celle mediante operatori che ne esprimono le relazioni; gli operatori possono essere operatori aritmetici, logici, funzionali.

3.2 I modi di riferimento ai valori delle celle: il riferimento in modo assoluto e in modo relativo

Il riferimento ai valori di celle singole può essere fatto in modo assoluto o in modo relativo. Con ciò si intende che nel primo caso si fa riferimento ad una data specifica cella, nel secondo ad una cella che dista da quella in cui si registra la formula un dato numero di righe e un dato numero di colonne.

In qualsiasi cella si intenda scrivere una formula il richiamo di una cella in modo assoluto viene indicato, se si fa riferimento alla cella di riga 1, colonna 1, con R1C1; se siamo nella riga 10 colonna 10 il richiamo in modo relativo del valore della cella di riga 1 colonna 1 viene indicato con R[-9]C[-9].

Il riferimento a celle in modo relativo è particolarmente utile in tutti quei casi una stessa formula, da impiegarsi in righe successive di una data colonna, non deve fare riferimento ad una cella di una data riga, ma a celle di righe successiva.

3.3 Gli operatori aritmetici.

Gli operatori aritmetici che si possono utilizzare sono:

^ elevazione a potenza
* moltiplicazione
/ divisione
% calcolo della percentuale
+ somma
- sottrazione.

3.4 Gli operatori logici

Gli operatori logici sono:

< maggiore di
<= maggiore, uguale a
> minore di
>= minore, uguale a
= uguale a
< > diverso

IF(condizione,se vero, se falso)
Esempio:

IF(R1C1>R3C1, R1C2, "non è maggiore")

in questo caso se R1C1 è maggiore di R3C1 alla cella della formula viene assegnato il valore R1C2, altrimenti viene visualizzato "non è maggiore".

AND(condizione 1, condizione 2,...condizione n). Si usa con IF.
Esempio:

IF(AND(R1C1<100, R2C1<100, R3C1>R1C1), "tutto verificato", "no")

OR(condizione1, condizione2,...condizione n). Si usa con IF.
Esempio:

IF(OR(R5C1<R5C2,R5C3<R5C4),R5C1,R10C10)

NOT(condizione) Si usa in congiunzione con IF.
Esempio:

IF(NOT(CONTO<=0),"SCOPERTO", "In regola")

3.5 I principali operatori funzionali

Numerosi sono gli operatori funzionali; si citano tra i più importanti in m 3.1:

ABS(N)	Da' il valore assoluto di N
AVERAGE(lista di celle)	Calcola la media della lista di celle
COUNT(lista di celle)	Conta il numero delle celle piene
LOOKUP(valore,tabella)	Ricerca in una tabella il valore corri spondente a quello dato (vedasi es.n.)
INT(N)	Da' l'intero di N
MAX(lista di celle)	Da' il massimo valore di cella
MIN(lista di celle)	Da' il minimo
MID(lista di celle)	Da' la media dei valori di cella
MOD(dividendo,divisore)	Da' il resto della divisione
NPV(r,lista di celle)	Da' il valore attuale al tasso r della lista di celle
ROUND(N,numero di cifre)	Da' l'arrotondamento di N
STDEV(lista di celle)	Da' la deviation standard delle celle
SIGN(N)	Da' il segno di N
SQRT(N)	Da' la radice quadrata di N
SUM(lista di celle)	Da' la somma dei valori di cella

Di funzioni importanti per la realizzazione di iterazioni di calcolo si parlerà negli esempi.

CAPITOLO 4

IL FOGLIO DI LAVORO ELETTRONICO

4.1 Come si prepara un foglio elettronico
4.2 I moduli base.
4.3 Moduli per la rappresentazione di serie temporali.
4.4 Moduli contabili
4.5 La concatenazione di più moduli

4.1 Come si prepara un foglio elettronico

Predisponendoci a costruire un foglio di lavoro elettronico occorre innanzitutto pulire lo schermo e la memoria da precedenti programmi; per ottenere ciò è necessario azionare il comando TRANSFERT CLEAR e premere la Y per confermare l'opzione. In questo modo la finestra sul foglio apparirà completamente libera, il cursore si troverà posizionato nella cella con coordinate R1C1 e la linea dei comandi sarà pronta a ricevere le istruzioni.

Si ritiene quindi che la preparazione di una tavola debba svolgersi nelle fasi seguenti:

1) preparazione del formato di visualizzazione dei dati;

2) registrazione delle formule ;

3) registrazione dei dati di prova;

4) controllo, correzione e blocco delle formule e dei titoli;

5) cancellazione dei dati di prova;

6) salvataggio del foglio in bianco;

7) registrazione dei dati effettivi;

8) stampa;

9) salvataggio dei dati effettivi.

Nella preparazione del formato di visualizzazione o stampa è spesso opportuno predisporre, oltre al foglio di output vero e proprio, anche un'area di lavoro per i calcoli e i risultati intermedi.

E' quindi opportuno provvedere al blocco delle formule e dei titoli per impedire che successive operazioni di cancellazione le possano eliminare.

E' bene infine provvedere al salvataggio di un foglio in bianco, questo foglio in bianco consente di effettuare la registrazione di nuovi dati senza operazioni di cancellazione di dati vecchi.

4.2 Moduli base.

Tutte le tabelle essendo "tabelle a doppia entrata" ammettono sia la lettura verticale che quella orizzontale. Si nota però che le tabelle per la rappresentazione di serie temporali hanno in prevalenza la distribuzione dei dati in senso orizzontale, mentre le tavole di calcolo hanno in prevalenza una distribuzione dei dati verticale. Le tavole delle serie statistiche temporali hanno infatti spesso i dati distribuiti orizzontalmente, pur ammettendo calcoli anche in senso verticale. Invece , le fatture, i conti correnti, i prospetti di redditività etc. richiedono quasi esclusivamente una lettura verticale.

Ciò è dovuto al fatto che originariamente lettura e scrittura vengono fatte per linee orizzontali, mentre i calcoli manuali richiedono l'incolonnamento verticale.

4.3 Moduli per la rappresentazione di serie temporali.

Uno dei tipi più comuni di tabella tabelle che si presta a svariate applicazioni è quello che serve alla rappresentazione di serie temporali.

In questa tabella abbiamo:

- nella prima colonna le voci di descrizione di ogni riga della tabella
- nelle colonne successive i valori di ogni scadenza temporale (annuale, mensile, giornaliera,...)
- nell'ultima colonna i totali per riga
- nell'ultima riga i totali per colonna

Una tabella con le suddette caratteristiche unisce ad un'estrema semplicità di realizzazione una grande versatilità d'applicazione; essa si presenta con la forma seguente:

	1983	1984	1985	TOT.
VENDITE 1	...	...	...	...
VENDITE 2	...	...	...	...
.....	...	...	...	...
TOTALE	....	....	....	....

Figura 4.3.1

Istruzioni di formattazione sono necessarie per fissare i titoli di colonna e di riga in modo da avere detti titoli sempre in vista anche quando si intende far scorrere le righe o le colonne.

Per fissare i titoli della tabella precedente si debbono usare i seguenti comandi:

```
|                                                                    |
|  WINDOW SPLIT TITLES: #of rows: 2  #of columns: 1                  |
|  Enter a number                                                    |
|                                                                    |
|  R1C1                        99% Free      Multiplan: TEMP         |
----------------------------------------------------------------------
```

Le istruzioni di calcolo che devono essere date per ottenere la suddetta tabella si riducono al calcolo dei totali per riga e per colonna: nel primo caso si tratterà di sommare gli importi di una stessa riga ma di colonne diverse, mentre nel secondo caso si dovranno sommare i diversi importi di una stessa colonna. Supponiamo di voler calcolare il totale della prima riga: occorre posizionare il cursore nella casella in cui si vuole il risultato, quindi si seleziona il comando *value* per preparare il programma ad accogliere una formula che potrà essere:

SUM(RC(-3):RC(-1))

o

RC(-3)+RC(-2)+RC(-1)

Qualora si intenda calcolare il totale per colonna avremo:

SUM(R(-3)C:R(-1)C)

o

R(-3)C+R(-2)C+R(-1)C

Inserite queste formule nelle celle in cui si vogliono i totali , al variare dei valori della tabella verranno ricalcolati automaticamente anche i totali che ricompariranno modificati.

E' possibile ottenere tabelle più complesse costruendo formule più articolate e in grado di eseguire diverse operazioni contemporaneamente. Vediamo ad esempio una tabella che rappresenta le vendite (espresse in milioni di lire) di un prodotto divise per aree geografiche durante il primo trimestre dell'anno: di queste si vuol conoscere oltre ai valori assoluti i valori percentuali, i totali progressivi e gli scostamenti dagli obiettivi prefissati per il trimestre.

ZONA	%	GEN	FEB	MAR	TOT	OBIET	DIFF
NORD	34.35	20	19	17	56	55	+1
CENTRO	31.90	17	17	18	52	54	-2
SUD	33.74	18	18	19	55	55	-
TOTALE	100	55	54	54	163	164	-1
PROGRESS.	...	55	109	163	326	...	...

Figura 4.3.2

Avendo gia illustrato le formule impiegate per calcolare i totali per riga e colonna soffermiamoci invece sulle formule per il calcolo delle percentuali, delle differenze e dei totali progressivi. Esse sono rispettivamente:

per le percentuali: RC(+4)*100/R(+4)C(+4)

per le differenze: RC(-2)-RC(-1)

per i totali progressivi: RC(-1)+R(-1)C

Si fa notare che nel MULTIPLAN IBM o TEXAS le parentesi con i valori di riferimento delle righe e delle colonne non sono tonde ma quadre.

4.4 Moduli contabili.

La pratica aziendale e finanziaria offre una grande quantità di esempi di moduli a sviluppo verticale. Hanno questo sviluppo: il Conto Economico a sezioni contrapposte, lo Stato Patrimoniale, il Conto Profitti e Perdite in forma progressiva, le fatture, i conti correnti, etc... In tutte queste tabelle il risultato finale ricercato compare nell'ultima riga del prospetto anche se talvolta sono necessari calcoli tra dati orizzontali per la determinazione di risultati parziali. Prendiamo in considerazione il prospetto di calcolo del margine netto di un prodotto di un'azienda che opera su due mercati in un mese della sua attività produttiva.

ARBI S.P.A. MARGINE	PRODOTTO 10		GENNAIO 1982 TOTALI PROD 10
descrizioni	MERCATI 11	12	
VENDITE	60.0	30.0	90.0
-costo ind. var.venduto	24.0	12.0	36.0
M.C. INDUSTRIALE	36.0	18.0	54.0
-costi var.commerciali	4.0	2.0	6.0
M.C. COMMERCIALE	32.0	16.0	48.0
-Spese fisse Commerc.	3.0	1.0	4.0
M.C.NETTO COMMERCIALE	29.0	15.0	44.0
Spese fisse industriali			6.0
MARGINE DI CONTRIBUZIONE NETTO DI PRODOTTO			38.0

ARBI S.P.A. MARGINE	PRODOTTO 20		GENNAIO 1984 TOTALI PROD 20
descrizioni	MERCATI 21	22	
VENDITE	40.0	25.0	65.0
-costo ind. var.venduto	17.0	10.0	27.0
M.C. INDUSTRIALE	23.0	15.0	38.0
-costi var.commerciali	1.0	0.5	1.5
M.C. COMMERCIALE	22.0	14.5	36.5
-Spese fisse Commerc.	2.0	1.0	3.0
M.C.NETTO COMMERCIALE	20.0	13.5	33.5
Spese fisse industriali			3.0
MARGINE DI CONTRIBUZIONE NETTO DI PRODOTTO			30.5

ARBI S.P.A. MARGINE	PRODOTTO 30			GENNAIO 1984 TOTALI PROD 30
descrizioni	MERCATI 31	32	33	
VENDITE	50.0	60.0	30.0	140.0
-costo ind. var.venduto	19.0	23.0	11.0	53.0
M.C. INDUSTRIALE	31.0	37.0	19.0	87.0
-costi var.commerciali	2.0	2.5	1.0	5.5
M.C. COMMERCIALE	29.0	34.5	18.0	81.5
-Spese fisse Commerc.	2.0	1.0	1.0	4.0
M.C.NETTO COMMERCIALE	27.0	33.5	17.0	77.5
Spese fisse industriali				4.0
MARGINE DI CONTRIBUZIONE NETTO DI PRODOTTO				73.5

Figura m 4.4.1 (segue)

ARBI S.P.A. MARGINE	PRODOTTO 40			GENNAIO 1984 TOTALI PROD 40
descrizioni	MERCATI 41	42	43	
VENDITE	35.0	30.0	40.0	105.0
-costo ind. var.venduto	12.0	10.0	13.0	35.0
M.C. INDUSTRIALE	23.0	20.0	27.0	70.0
-costi var.commerciali	2.0	1.0	1.8	4.8
M.C. COMMERCIALE	21.0	19.0	25.2	65.2
-Spese fisse Commerc.	3.0	1.0	1.0	5.0
M.C.NETTO COMMERCIALE	18.0	18.0	24.2	60.2
Spese fisse industriali				4.0
MARGINE DI CONTRIBUZIONE NETTO DI PRODOTTO				56.2

ARBI S.P.A. MARGINE	PRODOTTO 50		GENNAIO 1982 TOTALI PROD 50
descrizioni	MERCATI 51	52	
VENDITE	70.0	40.0	110.0
-costo ind. var.venduto	28.0	16.0	44.0
M.C. INDUSTRIALE	42.0	24.0	66.0
-costi var.commerciali	2.0	1.5	3.5
M.C. COMMERCIALE	40.0	22.5	62.5
-Spese fisse Commerc.	3.0	1.0	4.0
M.C.NETTO COMMERCIALE	37.0	21.5	58.5
Spese fisse industriali			7.0
MARGINE DI CONTRIBUZIONE NETTO DI PRODOTTO			51.5

ARBI S.P.A. MARGINE	PRODOTTO 60		GENNAIO 1984 TOTALI PROD 60
descrizioni	MERCATI 61	62	
VENDITE	60.0	35.0	95.0
-costo ind. var.venduto	17.0	10.0	27.0
M.C. INDUSTRIALE	43.0	25.0	68.0
-costi var.commerciali	4.0	3.0	7.0
M.C. COMMERCIALE	39.0	22.0	61.0
-Spese fisse Commerc.	2.0	1.0	3.0
M.C.NETTO COMMERCIALE	37.0	21.0	58.0
Spese fisse industriali			6.0
MARGINE DI CONTRIBUZIONE NETTO DI PRODOTTO			52.0

Figura m 4.4.1 (seguito)

ARBI S.P.A. MARGINE	PRODOTTO 70			GENNAIO 1984 TOTALI PROD 70
descrizioni	MERCATI 71	72	73	
VENDITE	55.0	65.0	50.0	170.0
-costo ind. var.venduto	19.0	24.0	16.0	59.0
M.C. INDUSTRIALE	36.0	41.0	34.0	111.0
-costi var.commerciali	3.0	3.5	2.0	8.5
M.C. COMMERCIALE	33.0	37.5	32.0	102.5
-Spese fisse Commerc.	2.0	2.0	1.5	5.5
M.C.NETTO COMMERCIALE	31.0	35.5	30.5	97.0
Spese fisse industriali				8.0
MARGINE DI CONTRIBUZIONE NETTO DI PRODOTTO				89.0

Figura m 4.4.1 (seguito)

La tecnica di calcolo del margine netto consiste, per ogni singolo mercato, di una serie progressiva di sottrazioni dalle vendite del prodotto per giungere a risultati netti di mercato e a somme orizzontali dei valori di ogni singolo mercato per arrivare al margine netto di prodotto.

ARBI S.P.A.	MARGINE DI STABILIMENTO SECCHIA				
descrizione	prodotti 10	20	30	40	TOT.
VENDITE	90.0	65.0	140.0	105.0	400.0
-Costo ind. var. del venduto	36.0	27.0	53.0	35.0	151.0
M. C. INDUSTRIALE	54.0	38.0	87.0	70.0	249.0
-Costi var. commerciali	6.0	1.5	5.5	4.8	17.8
M. C. COMMERCIALE	48.0	36.5	81.5	65.2	231.2
-Spese fisse commerciali	4.0	3.0	4.0	5.0	16.0
M. C. NETTO COMMERCIALE	44.0	33.5	77.5	60.2	215.2
-Spese fisse industrali	6.0	3.0	4.0	4.0	17.0
M. C. NETTO DI PRODOTTO	38.0	30.5	73.5	56.2	198.2
-Spese fisse di gruppo	0.0	7.0	15.0	10.0	32.0
M.C. NETTO PER GRUPPI		61.5	58.5	46.2	166.2
-Spese generali di stab.					50.0
M. C. NETTO DI STABILIMENTO					116.2

Figura m 4.4.2 (segue)

ARBI S.P.A.	CROSTOLO			
descrizione	prodotti			
	50	60	70	TOT.
VENDITE	110	95	170	375
-Costo ind. var. del venduto	44	27	59	130
M. C. INDUSTRIALE	66	68	111	245
-Costi var. commerciali	3.5	7	8.5	19
M. C. COMMERCIALE	62.5	61	102.5	226
-Spese fisse commerciali	4	3	5.5	12.5
M. C. NETTO COMMERCIALE	58.5	58	97	213.5
-Spese fisse industrali	7	6	8	21
M. C. NETTO DI PRODOTTO	51.5	52	89	192.5
-Spese fisse di gruppo	5		15	20
M.C. NETTO PER GRUPPI	46.5		126	172.5
-Spese generali di stab.				52
M. C. NETTO DI STABILIMENTO				120.5

Figura m 4.4.2 (seguito)

4.5 La concatenazione di più moduli

Talvolta per giungere a risultati più complessi è necessario utilizzare contemporaneamente più moduli così da sfruttare le relazioni logico-matematiche esistenti tra i dati di moduli diversi.

A ciò si perviene con la costruzione e il calcolo di tabelle tra loro concatenate, con gli output di alcune di esse che sono gli input di quelle successive, mediante l'impiego del comando EXTERNAL: Copy, List, Use.

Si prendano in esame, ad esempio, le tabelle di calcolo dei margini di mercato, di prodotto, di stabilimento e del risultato economico di una azienda (ARBIPP) che opera con due stabilimenti (SECCHIA e CROSTOLO) con più prodotti in più mercati.

Si consideri in primo luogo la tabella di calcolo del margine di contribuzione netto dei prodotti della ditta ARBI (figg. m 4.4.1) e si assegni a ciascuna colonna dei totali di prodotto un nome, (PROD10, PROD20, PROD30, etc.) con la funzione NAME; si salvi la tabella con il nome MARPROD.

Si passi quindi alla tabella di calcolo del margine di due stabilimenti (fig. m 4.4.2), ci si posizioni nella prima colonna in cui devono essere richiamati i totali dei margini di prodotto e si metta in esecuzione il comando EXTERNAL premendo il tasto X, quindi si dia il sottocomando Copy.

A questo punt apparirà sul monitor la richiesta del nome della tabella da cui un certo numero di righe colonne deve essere copiato, del nome dell'area da copiare e della opzione di LINK che consente di legare in modo permanente i dati della tabella dipendente con quelli della tabella di supporto.

Date le opportune risposte i totali dei margini di ciascun prodotto vengono letti nella tabella di supporto e visualizzati in quella dipendente.

Si attribuisca quindi a ciascuna colonna dei margini di stabilimento un nome quindi si salvi la tabella con il nome MARSTAB.

ARBI S.P.A.	CONTO ECONOMICO GENERALE	GENNAIO 1984	
	STABILIMENTI		
descrizione	SECCHIA	CROSTOLO	TOTALE
VENDITE	400.0	375.0	775.0
- Costo indust. var. venduto	151.0	130.0	281.0
M. C. INDUSTRIALE	249.0	245.0	494.0
- Costi variab. commerciali	17.8	19.0	36.8
M. C. COMMERCIALE	231.2	226.0	457.2
- Spese fisse specifiche	16.0	12.5	28.5
M. C. NETTO DI PRODOTTO	215.2	213.5	428.7
- Spese fisse di gruppo	17.0	21.0	38.0
M. C. NETTO COMMERCIALE	198.2	192.5	390.7
- Spese generali di stab.	32.0	20.0	52.0
M. C. NETTO DI STABILIMENTO	166.2	172.5	338.7
	50.0	52.0	102.0
	116.2	120.5	236.7
- Spese di periodo della sede		50.0	
- Spese di direzione tecnica		55.0	
- Spese commerciali		48.0	
- Spese amministrative		45.0	
- Oneri finanziari,Tributari,		15.0	
		213.0	213.0
UTILE NETTO			23.7

Figura m 4.5.1 .

Si passi alla tabella del conto economico generale e si richiamino i dati dei margini di stabilimento della tabella di supporto MARSTAB con il già visto comando EXTERNAL COPY.

Volendo visualizzare i nomi delle tabelle di supporto o dipendente da quella in uso si deve impiegare il comando EXTERNAL LIST; si otterranno i dati di cui alle figure che seguono:

```
Sheets supporting ARBIPP

MARSTAB


No sheets depend on ARBIPP



Press any key to redraw screen
R38C4                                   87% Free        Multiplan: ARBIPP
```

Figura m 4.5.2

```
Sheets supporting MARSTAB

MARPROD


Sheets depending on MARSTAB

ARBIPP




Press any key to redraw screen
R8C2     "VENDITE"                          84% Free      Multiplan: MARSTAB
```

Figura m 4.5.3

CAPITOLO 5

I PROBLEMI CARATTERISTICI

5.1 Il calcolo dei giorni intercorrenti tra due date.
Esempi di impiego:
5.1.1 scheda di deposito a risparmio (procedimento a NUMERI);
5.1.2 scheda di c/c di corrispondenza(procedimento amburghese a staffa.
5.2 La ricerca di un dato in una tavola di LOOKUP.
Esempi di impiego:
5.2.1 determinazione delle consistenze dei depositi e ricerca del tasso di interesse corrispondente;
5.3 L' ordinamento dei dati mediante la funzione SORT.
Esempio:
5.3.1 Il riordinamento delle operazioni per valuta nella staffa di un conto corrente.
5.4 La costruzione di modelli iterativi.
Esempio:
5.4.1 una catena di MARKOV per la determinazione delle quote attese di mercato.
5.5 Il trasferimento di dati dal DBASE al MULTIPLAN

5.1 Il calcolo dei giorni intercorrenti tra due date.

Un problema importante, preliminare rispetto a molti problemi finanziari di attualizzazione o calcolo del montante, è costituito dal calcolo del numero dei giorni intercorrenti tra due date. Normalmente si risolve questo problema calcolando, prima, il numero di giorni che intercorrono tra ciascuna data e una data fissa iniziale, poi, la differenza tra i due numeri trovati.

Il problema è complicato dal fatto che non tutti gli anni sono di 365 giorni, non tutti i mesi hanno lo stesso numero di giorni.

Il problema potrebbe essere risolto utilizzando una tabella con il numero dei giorni di ciascun mese e aggiungendo al mese di febbraio, normalmente di 28 giorni un giorno negli anni bisestili. Se è vero che il problema può essere risolto in questo modo, è anche vero che questa soluzione occuperebbe molto spazio per cui si ritiene opportuno seguire una tecnica diversa ricercando quella formula che dalla data consente di risalire direttamente al numero dei giorni che esistono tra di essa e il primo giorno di un dato anno, ad esempio il 1900.

Si nota che ogni quattro anni ne abbiamo uno bisestile di 366 giorni e che pertanto il numero dei giorni a partire dal 1900 è dato dal numero degli anni della data per 365,25.

Si nota ancora che, se per ipotesi il mese di febbraio avesse 30 giorni, nei primi 7 mesi si alternerebbero mesi di 31 con mesi di 30 giorni per cui, in media, avremmo mesi di giorni 30,5714. Prendendo il valore intero del prodotto di 0,5714 per il numero del mese della data si ottiene il numero dei mesi da 31 giorni compreso quello della data stessa.

Siccome in un calcolo di giorni interessa il numero dei mesi da 31 che precedono quello della data, si moltiplicano i mesi della data per 30,57 e non per 30,5714 e si sottrae 30. Così facendo avremo al settimo mese un totale di 183 giorni interi a partire dall'inizio dell'anno più il numero dei giorni della data come è in realtà e non 184. Utilizzando la stessa frazione per i 5 mesi successivi faremo scattare un intero a partire dal mese di agosto un mese sì e uno no anche per questi mesi così come è in realtà.

A questo punto si accerta se l'anno è bisestile calcolandone il resto della divisione per quattro, dopo di che dal conteggio dei giorni complessivi a partire dall'inizio si toglierà 1 se l'anno è bisestile , 2 se è un anno normale.
Quindi si aggiungeranno i giorni della data e il totale rappresenterà il totale dei giorni a partire dal 1900.

La formula da utilizzare nelle celle destinate ad accogliere il numero dei giorni a partire dal 1900 è:

```
INT(MESE*30.57)+INT(365.25*ANNO-395.25)-IF(AND(MESE>2,
MOD(ANNO,4)=0),1,IF(MESE>2,2,0) )
```

Nella formula le celle vengono identificate mediante, l'impiego della funzione NAME, con dei nomi, negli esempi le celle sono identificate mediante indicazione di coordinate relative.

5.1.1 Scheda di deposito a risparmio (procedimento a numeri).

Si presenta la scheda di deposito e "l'area di lavoro".

La scheda di deposito corrisponde esattamente ad una normale scheda tenuta con procedimento "a numeri".

```
        1                2                3     4  5  6      7          8
 2
 3  ---------------------------------------------------------------------------
 4      BANCA        AGRICOLA
 5 SCHEDA DI DEPOSITO A RISPARMIO                    CAT  LIBERO
 6
 7     r=          CONSISTENZA                        AL PORTATORE
 8         5             1829000   scadenza31 12 85          r=          5
 9  ---------------------------------------------------------------------------
10       DATE           OPERAZIONI   IMPORTI|DD|MM|AA  | GIORNINUMERI/100
11  ---------------------------------------------------------------------------
12    210985            VERSAMENTO  2100000 22  9 85        100    2100000
13    111085           PRELEVAMENTO  -400000  8 10 85         84    -336000
14                                  ---------                        ---------
15                                    1700000                          1764000
16    151285            VERSAMENTO   500000 18 12 85         13      65000
17                                  ---------                        ---------
18                                    2200000                          1829000
19    311284      CAPITALIZZAZIONE     18791 31 12 85          0           0
20                                  ---------                        ---------
21                                    2218791                            *****
22                                          0  0  0  0
23                                  ---------                        ---------
```

Figura 5.1.1

In essa, nella colonna degli importi si è proceduto alla ripetizione della funzione somma ad ogni nuova operazione mediante il comando

COPY FROM ...TO ., ., .

che consente di copiare una formula in determinate caselle anche saltando qualche riga o qualche colonna.

I giorni sono per ogni operazione calcolati fino alla fine dell'anno per ogni operazione.

I numeri risultano dal prodotto (diviso 100) di giorni per importi. Le formule sono state copiate con il solito comando COPY FROM...TO....

```
            9                    10                    11
 2
 3 "__________________ "__________________ "__________________
   __________"          __________"          __________"
 4
 5                      "AREA DI "           "LAVORO"
 6
 7                      "GIORNI "            "DAL PRIMO"
 8                      IF(AND(RC[-5]>2,MOD  INT(RC[-6]*30.57)+I
                        (RC[-4],4)=0),-1,IF  NT(365.25*RC[-5]-39
                        (RC[-5]>2,-2,0))     5.25)+RC[-7]+RC[-1]
 9
10 " INTERESSI"
11
12 RC[-1]*R8C8*RC[-2]/  IF(AND(RC[-5]>2,MOD  INT(RC[-6]*30.57)+I
   36500                (RC[-4],4)=0),-1,IF  NT(365.25*RC[-5]-39
                        (RC[-5]>2,-2,0))     5.25)+RC[-7]+RC[-1]
13                      IF(AND(RC[-5]>2,MOD  INT(RC[-6]*30.57)+I
                        (RC[-4],4)=0),-1,IF  NT(365.25*RC[-5]-39
                        (RC[-5]>2,-2,0))     5.25)+RC[-7]+RC[-1]
14
15 RC[-1]*R8C8/365
16 RC[-1]C*0.75         IF(AND(RC[-5]>2,MOD  INT(RC[-6]*30.57)+I
                        (RC[-4],4)=0),-1,IF  NT(365.25*RC[-5]-39
                        (RC[-5]>2,-2,0))     5.25)+RC[-7]+RC[-1]
17
18 RC[-1]*R8C8/365
19 RC[-1]C*0.75         IF(AND(RC[-5]>2,MOD  INT(RC[-6]*30.57)+I
                        (RC[-4],4)=0),-1,IF  NT(365.25*RC[-5]-39
                        (RC[-5]>2,-2,0))     5.25)+RC[-7]+RC[-1]
20
21 RC[-1]*R8C8/365
22 RC[-1]C*0.75         IF(AND(RC[-5]>2,MOD  INT(RC[-6]*30.57)+I
                        (RC[-4],4)=0),-1,IF  NT(365.25*RC[-5]-39
                        (RC[-5]>2,-2,0))     5.25)+RC[-7]+RC[-1]
```

Figura m 5.1.2

```
                 10         11         12         13         14
 2
 3 ------------------------------------------------------------------||
 4                                                                  ||
 5              AREA DI     LAVORO                                  ||
 6                                                                  ||
 7               GIORNI  DAL PRIMO                                  ||
 8                   -2     31046                                   ||
 9 -                                                                ||
10  INTERESSI                      CONSISTENZA    TAVOLA DI LOOKUP||
11                                                   0          5 ||
12 28767.123         -2     30946              5000000        6.5 ||
13                   -2     30962             10000000        7.5 ||
14                                             15000000        8.5 ||
15 24164.384                         1764000  20000000         10 ||
16 18123.288         -2     31033                                  ||
17                                                                  ||
18 25054.795                         1829000                        ||
19 18791.096         -2     31046                                  ||
20                                                                  ||
```

Figura 5.1.3

L'area di lavoro consente di calcolare i giorni dalla data iniziale di ciascuna operazione e provvede al calcolo immediato di interessi e ritenute erariali nella misura del 25%. Il tasso di interesse è dato dall'operatore in riga 8 colonna 8.

Gli interessi vengono alla fine capitalizzati dall'operatore mediante registrazione di quelli risultanti in colonna 9 dell'area di lavoro.

Inoltre nell'area di lavoro è stata predisposta un'area per il calcolo delle consistenze ed una tavola di LOOKUP che, in rapporto alla consistenza registrata nella scheda, riga 8 colonna 2, calcola l'interesse corrispondente.

5.2 La ricerca di un dato in una tavola di LOOKUP.

La funzione di LOOKUP ricerca in una tavola di due colonne (o due righe) di valori corrispondenti quello che corrisponde ad un valore dato. Possiamo ad esempio avere in una colonna gli scaglioni di reddito e, nella colonna accanto, le corrispondenti aliquote di imposta. Dato il nostro reddito ci possiamo chiedere quale aliquota dobbiamo pagare. Allo spiacevole quesito risponde la funzione :

LOOKUP(Reddito,area di ricerca)

Se i valori tra cui possiamo trovare quello dato iniziano da R10C5 e finiscono in R20C5, e i valori corrispondenti sono in colonna 6, possiamo definire l'area di ricerca con R10C5:R20C6.

Trovato, a partire dal primo, un valore superiore a quello dato, la funzione restituisce il valore che nella colonna di fianco a destra gli corrisponde.

5.2.1 Determinazione delle consistenze dei depositi e ricerca del tasso di interesse corrispondente.

```
        9         10         11         12         13         14         15
 3 ------------------------------------------------------------------||
 4                                                                  ||
 5              AREA DI     LAVORO                                  ||
 6                                                                  ||
 7               GIORNI  DAL PRIMO                                  ||
 8                   -2     31046                                   ||
 9                                                                  ||
10  INTERESSI                      CONSISTENZA    TAVOLA DI LOOKUP||
11                                                   0          5 ||
12 82191.781         -2     30946              5000000        6.5 ||
13                   -2     30962             10000000        7.5 ||
14                                            15000000        8.5 ||
15 77013.699                          5622000 20000000         10 ||
16 57760.274         -2     31033                                  ||
17                                                                  ||
18 78082.192                          5700000                       ||
19 58561.644         -2     31046                                  ||
20                                                                  ||
21    #VALUE!                          #VALUE!                      ||
22    #VALUE!         0      -396                                   ||
```

Figura m 5.2.1

```
                    1
 8 LOOKUP(RC[+1],R[+3]
   C[+12]:R[+7]C[+13])
```

Figura m 5.2.2

```
                 12                    13                    14
 9
10 "CONSISTENZA"          "  TAVOLA"          "DI LOOKUP"
11                        0                   5
12                        5000000             6.5
13                        10000000            7.5
14                        15000000            8.5
15 RC[-4]*100/R12C7       20000000            10
16
17
18 RC[-4]*100/R12C7
```

Figura m 5.2.3

```
10 " INTERESSI"
11
12 RC[-1]*R8C8*RC[-2]/36500
13 R[-1]C*0.75
14
15 RC[-1]*R8C8/365
16 R[-1]C*0.75
17
18 RC[-1]*R8C8/365
19 R[-1]C*0.75
20
21 RC[-1]*R8C8/365
22 R[-1]C*0.75
```

```
12 R[-4]C[+4]-RC[+4]
13 IF(RC[-1]>0,R8C11-RC[+4]," ")
14
15
16 IF(RC[-1]>0,R8C11-RC[+4]," ")
17
18
19 IF(RC[-1]>0,R8C11-RC[+4]," ")
```

Figura m 5.2.4

```
                    10                                          11
 7 "GIORNI "                                    "DAL PRIMO"
 8 IF(AND(RC[-5]>2,MOD(RC[-4],4)                INT(RC[-6]*30.57)+INT(365.25*
   =0),-1,IF(RC[-5]>2,-2,0))                    RC[-5]-395.25)+RC[-7]+RC[-1]
 9
10
11
12 IF(AND(RC[-5]>2,MOD(RC[-4],4)                INT(RC[-6]*30.57)+INT(365.25*
   =0),-1,IF(RC[-5]>2,-2,0))                    RC[-5]-395.25)+RC[-7]+RC[-1]
13 IF(AND(RC[-5]>2,MOD(RC[-4],4)                INT(RC[-6]*30.57)+INT(365.25*
   =0),-1,IF(RC[-5]>2,-2,0))                    RC[-5]-395.25)+RC[-7]+RC[-1]
14
15
16 IF(AND(RC[-5]>2,MOD(RC[-4],4)                INT(RC[-6]*30.57)+INT(365.25*
   =0),-1,IF(RC[-5]>2,-2,0))                    RC[-5]-395.25)+RC[-7]+RC[-1]
17
18
19 IF(AND(RC[-5]>2,MOD(RC[-4],4)                INT(RC[-6]*30.57)+INT(365.25*
   =0),-1,IF(RC[-5]>2,-2,0))                    RC[-5]-395.25)+RC[-7]+RC[-1]
20
21
22 IF(AND(RC[-5]>2,MOD(RC[-4],4)                INT(RC[-6]*30.57)+INT(365.25*
   =0),-1,IF(RC[-5]>2,-2,0))                    RC[-5]-395.25)+RC[-7]+RC[-1]
```

Figura m 5.2.5

Nella nostra scheda di deposito un'area di LOOKUP viene richiamata da riga 8 colonna 1 per cercare i tassi corrispondenti alla giacenza registrata dall'operatore in riga 8 colonna 2, sulla base dei dati forniti dal programma in colonna 12.

La formula di riga 8 colonna 1 è:

LOOKUP(RC[+1],R[+3]C[+12]:R[+7] [+13])

Con detta formula si ricerca il valore corrispondente a RC[+1] nella colonna C[+12] e si restituisce il corrispondente in colonna C[+13].

5.3 L'ordinamento dei dati mediante la funzione SORT

Con questa funzione è possibile riordinare tutte o un gruppo di righe di un foglio elettronico, in ordine crescente o decrescente, sulla base dei valori di una data colonna. Per il riordino è pertanto necessario dichiarare la colonna di riordino, la prima e l'ultima riga dell'area da riordinare e il senso, crescente o decrescente, del riordino voluto. Il riordino può essere eseguito sia sulla base di numeri che su testi.

5.3.1 Il riordinamento delle operazioni per valuta in una staffa di un conto corrente.

In un conto corrente di corrispondenza i giorni di interesse si calcolano sui saldi per valuta da una operazione alla successiva. E' pertanto necessario che le operazioni registrate in ordine di data nella scheda vengano riordinate in ordine di valuta nella staffa di conteggio interessi.

Per fare ciò è necessario riservare un gruppo di righe alla scheda per la registrazione delle operazioni man mano si succedono (fig. m 5.3.1), ed un gruppo di righe sotto alle prime per la staffa.

Nel momento in cui si vogliono conteggiare gli interessi si deve procedere al SORT dei dati della scheda sulla base della colonna in cui sono calcolati i giorni a partire da un anno iniziale (m 5.3.2).

A questo punto si copiano con il comando COPY le righe della scheda, con importi e valute nella staffa (m 5.3.3).

Quindi si da il comando di ricalcolo per ottenere i **numeri** e gli interessi e mettere in esecuzione la parte di foglio elettronico destinata al prospetto delle SPESE E COMPETENZE (m 5.3.4) e(m 5.3.5).

Le formule della staffa si ritrovano nei prospetti m 5.3.6, m 5.3.7, m 5.3.8 e m 5.3.9.

CASSA DI RISPARMIO DI REGGIO EMILIA
Filiale di Casalgrande

Scheda del c/c di corrispondenza n. 3256
intestato a Anselmo Rampoldi ,Via Roma 3, Casalgrande.

DATE	OPERAZIONI	IMPORTI DARE	IMPORTI AVERE	VALUTE DD	VALUTE MM	VALUTE AA	S	GIORNI
1/10/84	Residuo credito		6750000	30	9	84		30985
4/10/84	Versamento		480000	5	10	84		30990
11/10/84	Saldo competenze		465380	30	9	84		30985
27/10/84	Bonifico stip.		1255730	27	10	84		31012
6/11/84	A/B n. 3256	850000		25	10	84		31010
18/11/84	Prelevamento ATR	370000		18	11	84		31034
27/11/84	Bonifico stip.		1255730	27	11	84		31043
1/12/84	Giro vs. affitto		930000	2	12	84		31048
8/12/84	A/B n. 3257	670500		26	11	84		31042
19/12/84	Pagamento SIP	95760		19	12	84		31065
27/12/84	Bonifico stip.		1257800	27	12	84		31073
31/12/84	Vs. competenze							0
								0
								0
								0
								0

Figura m 5.3.1

	1	2	3	4	5	6	7	8	9
29		Dopo la copia da R14C3:R25C7							0
30									
31		NUMERI	IMPORTI		VALUTE				SALDI
32	DARE	AVERE	DARE	AVERE	DD	MM	AA	S	IMPORTI
33									
34				6750000	30	9	84	D	0
35				465380	30	9	84	D	0
36				480000	5	10	84	D	0
37			850000		25	10	84	D	0
38				1255730	27	10	84	D	0
39			370000		18	11	84	D	0
40			670500		26	11	84	D	0
41				1255730	27	11	84	D	0
42				930000	2	12	84	D	0
43			95760		19	12	84	D	0
44				1257800	27	12	84	D	0
45					31	12	84	D	0

Figura m 5.3.2

	1	2	3	4	5	6	7	8	9
9	Dopo il Sort								
10									
11		OPERAZIONI	IMPORTI		VALUTE				
12	DATE		DARE	AVERE	DD	MM	AA	S	GIORNI
13									
14	1/10/84	Residuo credito		6750000	30	9	84		30985
15	11/10/84	Saldo competenze		465380	30	9	84		30985
16	4/10/84	Versamento		480000	5	10	84		30990
17	6/11/84	A/B n. 3256	850000		25	10	84		31010
18	27/10/84	Bonifico stip.		1255730	27	10	84		31012
19	18/11/84	Prelevamento ATR	370000		18	11	84		31034
20	8/12/84	A/B n. 3257	670500		26	11	84		31042
21	27/11/84	Bonifico stip.		1255730	27	11	84		31043
22	1/12/84	Giro vs. affitto		930000	2	12	84		31048
23	19/12/84	Pagamento SIP	95760		19	12	84		31065
24	27/12/84	Bonifico stip.		1257800	27	12	84		31073
25	31/12/84	Vs. competenze			31	12	84		0
26									0
27									0

Figura m 5.3.3

	1	2	3	4	5	6	7	8	9	10
28									0	
29	Dopo l'ordine di ricalcolo								0	
30										
31	NUMERI		IMPORTI		VALUTE				SALDI	
32	DARE	AVERE	DARE	AVERE	DD	MM	AA	S	IMPORTI	dd
33										
34				6750000	30	9	84	A	6750000	0
35		36076900		465380	30	9	84	A	7215380	5
36		153907600		480000	5	10	84	A	7695380	20
37		13690760	850000		25	10	84	A	6845380	2
38		178224420		1255730	27	10	84	A	8101110	22
39		61848880	370000		18	11	84	A	7731110	8
40		7060610	670500		26	11	84	A	7060610	1
41		41581700		1255730	27	11	84	A	8316340	5
42		157187780		930000	2	12	84	A	9246340	17
43		73204640	95760		19	12	84	A	9150580	8
44		41633520		1257800	27	12	84	A	10408380	4
45					31	12	84	D	0	###
46								*	0	0
47								*	0	0
48								*	0	0
49								*		
50										
51		764416810	1986260	12394640	VS			A	10408380	
52										

Figura m 5.3.4

```
|-----------------------------------------------------------------------
|          PROSPETTO SPESE E COMPETENZE AL
| Egregio Signor xxxxxx  xxxx xxxx  via xxxxx
|                                    tasso     8 %
|Vs Avere:
|
|interessi        764416810      *          8
|                 ------------------------------ =          167543
|                              36500
|
| ritenuta fiscale 25 %       167543     *    25              41886
|                            ---------------------=        ----------
|                                        100    Vs.AVERE    209429
|
|Vs Dare:
|         spese tenuta conto       L.     500 N.R.  15         7500
|
|         spese generali           L.    5000                  5000
|                                                          ----------
|                                                Vs. DARE     12500
|
|   VOSTRO   AVERE a saldo    196929 valuta   31 12 85
|
|-----------------------------------------------------------------------
```

Figura m 5.3.5

```
                                                  1
49 "Calcolo della sommatoria dei"
50 "NUMERI DARE"
51 SUM(R[-17]C:R[-2]C)


                                                  2
49 "Calcolo della sommatoria dei"
50 "NUMERI AVERE"
51 SUM(R[-17]C:R[-2]C)


                                                  3
49 "Calcolo della sommatoria degli"
50 "importi DARE"
51 SUM(R[-17]C:R[-2]C)


                                                  4
49 "Calcolo della sommatoria degli"
50 "importi AVERE"
51 SUM(R[-17]C:R[-2]C)
```

Figura m 5.3.6

```
                                        9
30
31 "Formula per il calcolo dei"
32 "Saldi"
33
34 IF(RC[-5]>0,RC[-5],RC[-6])
```

```
                                        2
30
31 "Formula per il calcolo dei"
32 "Numeri Avere"
33
34 IF(RC[+8]>0,RC[+7]*RC[+8],"|                    |")
```

```
                                       10
30
31 "Formula per il calcolo dei"
32 "  giorni tra due valute"
33
34 IF(R[+1]C[+1]>=0,R[+1]C[+1]-RC[+1],0)
```

Figura m 5.3.7

```
                                       11
30
31 "Formula per il calcolo dei "
32 "giorni dalla valuta all'anno I"
33
34 INT(RC[-5]*30.57)+INT(365.25*RC[-4]-395.25)+RC[-6]+(IF(AND(
   RC[-5]>2,MOD(RC[-4],4)=0),-1,IF(RC[-5]>2,-2,0)))+396
```

Figura m 5.3.8

```
                                        9
10
11 "Formule per il calcolo dei"
12 "giorni dall'anno I"
13
14 INT(RC[-3]*30.57)+INT(365.25*RC[-2]-395.25)+RC[-4]+(IF(AND(
   RC[-3]>2,MOD(RC[-2],4)=0),-1,IF(RC[-3]>2,-2,0)))+396
```

Figura m 5.3.9

5.4 La costruzione di modelli iterativi

Si consideri il sistema di equazioni di primo grado x=1000 y=10/100*z e z=x-y .

Normalmente la soluzione di questo sistema si può ottenere applicando il metodo di sostituzione, o di confronto,o di addizione e sottrazione, o della combinazione lineare, o il metodo di KRAMER.

Con MULTIPLAN è possibile risolvere il problema ricorrendo al metodo iterativo per il quale il programma partendo dal valore assegnato alla prima equazione, risolve la seconda, poi la terza e di nuovo la seconda, fino a quando la variazione dei valori delle singole variabili siano più piccole di un valore DELTA definito dalla macchina o dall'operatore con apposita istruzione.

La successione dei valori delle variabili sarà:

X	Y	Z
1000	0	1000
	100	900
	90	910
	91	909

etc. fino a quando $Y_n - Y_{n-}$ e $Z_n - Z_{n-1}$ non siano minori di DELTA.

Per la esecuzione delle iterazioni MULTIPLAN mette a disposizione il comando:

```
|                                                                    |
|   OPTION    recalc: Yes  No        mute: Yes No                    |
|             iteration : Yes  No    completion test at :            |
|   Select option                                                    |
|    R1C1                          99% Free     Multiplan: Temp      |
|--------------------------------------------------------------------|
```

Bloccato il ricalcolo e l'avviso sonoro si deve scegliere Yes per le iterazioni, quindi dare l'ordine di ricalcolo con "shift".

Per il controllo delle iterazioni Multiplan mette a disposizione dell'operatore nello stesso comando OPTION un test di completamento ed inoltre il seguente gruppo di funzioni:

ITERCNT()> numero	Da' TRUE se le iterazioni sono maggiori di "numero".
DELTA()< numero	da' TRUE quando i risultati mutano per meno di "numero"
IF(ISNA(ITERCNT()), iniziale,formula)	definisce il valore iniziale di una casella per la prima iterazione.

Mettendo il test di completamento alla casella in cui si è registrata la funzione DELTA() o quella di ITERCNT(), nel momento in cui i valori relativi passano da FALSE a TRUE le iterazioni si arrestano.

Si nota che la funzione ITERCNT()(e DELTA() possono essere usate in alternativa consentendo di definire l'una il numero delle iterazioni, l'altra un valore massimo sotto il quale le iterazioni si arrestano.

Si fa notare che l'ultima formula può essere utilizzata per costruire un contatore di iterazioni. Infatti non si può sommare un valore a quello di una cella se in essa non vi è un valore iniziale. Pertanto si deve nella cella in questione scrivere la formula seguente o un'altra equivalente:

IF(ISNA(ITERCNT()),0,RC+1)

Il metodo delle iterazioni può essere utilizzato anche per sistemi con un grande numero di equazioni che richiedono di solito sofisticate tecniche di soluzione. Si pensi almeno al problema del calcolo del tasso interno di rendimento, a molti problemi di calcolo di costi indiretti etc..

5.4.1 Una catena di MARKOV per la determinazione delle quote attese di mercato

Supponiamo che una azienda, che produce 4 tipi di succo di frutta, intenda lanciarne un altro e si chieda quale sarà la struttura del mercato dei suoi prodotti dopo questo lancio. Oppure si supponga che una azienda, accertate le probabilità di passaggio di un consumatore da una marca all'altra, intenda prevedere la futura struttura del mercato.

Si disponga per ciascun prodotto dei dati delle probabilità di permanenza del consumatore o di passaggio alle altre marche e si distribuiscano dette probabilità nella seguente matrice:

	P.1	P.2	P.3	P.4	P.5
P.1 UVA NERA	0.6	0.06	0.15	0.2	0.02
P.2 F.ESOTICI	0.05	0.4	0.3	0.2	0.08
P.3 UVA BIANCA	0.2	0.25	0.4	0.25	0.05
P.4 ANANAS	0.15	0.02	0.1	0.3	0.2
P.5 POMPELMO	0.1	0.3	0.05	0.05	0.05

Figura m 5.4.1

Si disponga inoltre dei dati delle attuali quote di mercato che sono:

P.1 UVA NERA	0.15
P.2 F.ESOTICI	0.2
P.3 UVA BIANCA	0.05
P.4 ANANAS	0.35
P.5 Pompelmo	0.25

Figura m 5.4.2

La determinazione delle quote attese di mercato richiede la soluzione simultanea del sistema delle cinque equazioni delle quote di mercato più una equazione per il totale complessivo delle quote:

$$(P.1)_{t+1} = 0.6*(P.1)_t + 0.05*(P.2)_t + 0.2*(P.3)_t + 0.15*(P.4)_t + 0.1*(P.5)_t$$
$$(P.2)_{t+1} = 0.06*(P.1)_t + 0.4*(P.2)_t + 0.25*(P.3)_t + 0.02*(P.4)_t + 0.3*(P.5)_t$$
$$(P.3)_{t+1} = 0.15*(P.1)_t + 0.3*(P.2)_t + \ldots\ldots + \ldots\ldots + \ldots\ldots$$
$$(P.4)_{t+1} = 0.2*(P.1)_t + 0.2*(P.2)_t + \ldots\ldots + \ldots\ldots + \ldots\ldots$$
$$(P.5)_{t+1} = 0.02*(P.1)_t + 0.08*(P.2)_t + \ldots\ldots + \ldots\ldots + \ldots\ldots$$
$$P.1+P.2+P.3+P.4+P.5 = 1$$

Figura m 5.4.3

1	2	3	4	5	6	7	8	9	10	11
CATENE DI	MARCOV									
	QUOTA	TRUE			MATRICE DI TRANSIZIONE					
PRODOTTI	DI MERCATO	INIZIALE	FINALE		P. 1	P. 2	P. 3	P. 4	P. 5	
P.1= UVA NERA	0.15	0.2424162	0.2424162	||	0.6	0.06	0.15	0.2	0.02	||
P.2=F.ESOTICI	0.2	0.1925824	0.1925824	||	0.05	0.4	0.3	0.2	0.08	||
P.3=UVABIANCA	0.05	0.2042993	0.2042993	||	0.2	0.25	0.4	0.25	0.05	||
P.4=ANANAS	0.35	0.2081462	0.2081462	||	0.15	0.02	0.1	0.3	0.2	||
P.5= POMPELMO	0.25	0.1525558	0.148377	||	0.1	0.3	0.05	0.05	0.5	||
TOTALE	1	1	0.9958212						TRUE	||

Figura m 5.4.4

La soluzione di questo sistema può essere ottenuta in molti modi, ad esempio mediante il calcolo matriciale con l'inversione della matrice, in ogni caso si tratta di tecniche che richiedono discrete conoscenze di matematica. Proviamo con il metodo iterativo, vedremo che si richiedono conoscenze di matematica solo elementari. Infatti basta registrare il sistema con le relative funzioni di controllo delle iterazioni e di assegnazione dei valori iniziali per poterne chiedere la messa in esecuzione ed ottenerne la soluzione (m 5.4.6).

Le funzioni e le formule utilizzate in colonna 3 e in colonna 4 sono rispettivamente le seguenti:

```
                    3
 2
 3 DELTA()<0.000001
 4 " INIZIALE"
 5 "----------------------------------------"
 6 IF(ITERCNT()<40,RC[-1],RC[+1])
 7 IF(ITERCNT()<40,RC[-1],RC[+1])
 8 IF(ITERCNT()<40,RC[-1],RC[+1])
 9 IF(ITERCNT()<40,RC[-1],RC[+1])
10 1-SUM(R[-4]C:R[-1]C)
11 "----------------------------------------"
12 SUM(R[-6]C:R[-2]C)
```

Figura m 5.4.5

```
                                    4
 2
 3
 4 "FINALE"
 5 "----------------------------------------"
 6 RC[-1]*RC[+2]+R[+1]C[-1]*R[+1]C[+2]+R[+2]C[-1]*R[+2]C[+2]+R
   [+3]C[-1]*R[+3]C[+2]+R[+4]C[-1]*R[+4]C[+2]
 7 R[-1]C[-1]*R[-1]C[+3]+RC[-1]*RC[+3]+R[+1]C[-1]*R[+1]C[+3]+R
   [+2]C[-1]*R[+2]C[+3]+R[+3]C[-1]*R[+3]C[+3]
 8 RC[-1]*RC[+4]+R[+1]C[-1]*R[+1]C[+4]+R[+2]C[-1]*R[+2]C[+4]+R
   [-2]C[-1]*R[-2]C[+4]+R[-1]C[-1]*R[-1]C[+4]
 9 RC[-1]*RC[+5]+R[+1]C[-1]*R[+1]C[+5]+R[-3]C[-1]*R[-3]C[+5]+R
   [-2]C[-1]*R[-2]C[+5]+R[-1]C[-1]*R[-1]C[+5]
10 RC[-1]*RC[+6]+R[-4]C[-1]*R[-4]C[+6]+R[-3]C[-1]*R[-3]C[+6]+R
   [-2]C[-1]*R[-2]C[+6]+R[-1]C[-1]*R[-1]C[+6]
11 "----------------------------------------"
12 SUM(R[-6]C:R[-2]C)
```

Figura m 5.4.6

5.5 Il trasferimento di dati dal DBASE al MULTIPLAN

Può essere necessario trasferire dei dati da DBASE e MULTIPLAN o da MULTIPLAN a DBASE per poter sfruttare le diverse potenzialità dei due programmi. Ad esempio si immagini i dover elaborare i dati delle vendite per zona dei rappresentanti per poter ottenere delle rappresentazioni grafiche. Si dovranno riprendere i dati da DBASE per lavorati con il MULTIPLAN e quindi rappresentarli graficamente con GRAPHIC ASSISTANT o un altro similare programma di BUSINESS-GRAPHICS.

Tutti questi programmi hanno delle utility che consentono di trasformare i loro dati in

caratteri ASCII in formato leggibile da un EDITOR di sistema in caratteristici file sequenziali. Tutti questi programmi possono leggere file sequenziali con determinate caratteristiche.

Consideriamo in primo luogo i file leggibili da Multiplan.

Multiplan, oltre ai file prodotti da programma in fase di salvataggio senza opzioni, può produrre e leggere file in formato simbolico con nomi di estensione .SLK. o .TXT.

Detti file, che possono essere prodotti dall'utente a partire da un comune programma editore (EDLIN per gli utenti di MS.DOS), devono essere composti di record separati da un ritorno carrello e da campi separati da un ";".

Ciascun record avrà un primo campo che ne identifica il tipo e quindi il contenuto.

Il primo di questi record deve obbligatoriamente avere il seguente contenuto:

ID;PMP

PMP identifica il PROGRAMMA che metterà in esecuzione il file;

Il secondo di questo record deve definire l'area della tabella elettronica cui i dati del file debbono fare riferimento:

B;Yr;Xc

B è l'identificatore del tipo di record (appunto B), Y, X sono costanti obbligatorie, r, c sono i valori dei limiti estremi di riga e di colonna della tabella elettronica di riferimento.

In record di tipo F si definiscono le caratteristiche di formattazione della casella definita dal record stesso.

In record di tipo C si identificano riga e colonna e valore o formula di calcolo. Detti record hanno il seguente contenuto essenziale:

C;Yr;Xc;K

Come visto precedentemente Y e X sono costanti obbligatorie mentre r e c sono gli identificatori di casella. K è il valore contenuto nella casella.

L'ultimo record deve essere di tipo E. E' composto dalla sola lettera E.

Il file di registrazione di una colonna di descrizione delle zone di vendita e della colonna dei valori relativi sarà pertanto il seguente:

```
ID;PMP
B;X2;Y6
C;X1;Y1;K"EMILIA"
C;X1;Y2;K"ROMAGNA"
C;X1;Y2;K"VENETO"
C;X1;Y4;K"LOMBARDIA"
```

```
C;X1;Y5;K"TOSCANA"
C;X1;Y6;K"PIEMONTE"
C;X2;Y1;K15580900
C;X2;Y2;K12850500
C;X2;Y3;K15890200
C;X2;Y4;K13570900
C;X2;Y5;K12300000
C;X2;Y6;K14200800
E
```

Il file costruito con l'editor e salvato con un nome e l'estensione .SLK oppure .TXT, può essere messo in esecuzione da MULTIPLAN mediante il caricamento del file stesso dopo che sia stata scelta l'opzione $FF03 "symbolic" $FF01.

Spesso i dati da elaborare debbono essere ripresi da un file registrato da DBASE oppure da un programma BASIC o in altro linguaggio.
In questo caso si deve procedere nel modo seguente:

1) predisporre una struttura record con campi che possano accogliere i dati obbligatori di un file .SLK;

2) registrare i primi due record con l'identificatore di programma e l'area di riferimento della tabella elettronica;

3) appendere con apposito programma dal file di dati con estensione .DBF i dati da copiare e gli identificatori di riga e di colonna delle caselle MULTIPLAN;

4) copiare il file così preparato in un file formato sistema con estensione .TXT;

5) sostituire mediante l'editore automaticamente tutte le virgole con punti e virgola e aggiungere il record di chiusura E.

1) Struttura record

Ciascun record deve avere un minimo di 7 campi di cui quattro rispettivamente per l'indicatore di tipo record, la costante Y, la costante X, la costante K, e tre di cui uno per l'indicatore di riga, uno per l'indicatore di colonna e uno per il contenuto di casella. Nel nostro caso, dovendosi copiare due campi uno di nome "KD" l'altro di nome "KN", rispettivamente per le zone di vendita e i relativi valori, dal file di dati originari al file di formato .SLK, il numero dei campi sarà 8.

Sia VECCHIO.DBF il file di dati da trasferire per i campi KD, e KN.
Da DBASE con il comando CREATE diamo nome TRASF al file di deposito temporaneo dei dati da copiare. La sua struttura sia, coerentemente con quanto detto sopra la seguente:

CAMPO	NOME	TIPO	LARGHEZZA
1	TIPO	C	2
2	X	C	3
3	nx	N	3
4	Y	C	2
5	ny	N	3
6	K	C	2
7	KD	C	11
8	KN	N	9

2) Registrazione dei primi due record.

Con l'istruzione APPEND si registri nel primo record

ID PMP

e nel secondo

B Y 6 X 2

3) Il programma DB per apprendere i dati da trasferire al MULTIPLAN può essere simile al seguente:

```
***************** SALVATO COME TRAS.PRG ***********************************
** Questo programma consente di trasferire dati da nome.dbf  a nome.slk ****
** La struttura record di riferimento e' la seguente:                    **
** 1) ID,C,2  2) X,C,3  3) XN,N,3 4) Y,C,2  5) YN,N,3  6) K,C,2          **
** 7) KD,C,11 8) KN,N,9                                                  **
** I campi 1,2,4 e 6 hanno contenuto COSTANTE                            **
** I campi 3 (XN) e 5 (YN) contengono il numero di riga e di colonna delle**
** caselle di MULTIPLAN                                                  **
** I campi 8 e 9 contengono la "descrizione e il valore" delle caselle di **
** MULTIPLAN                                                             **
** Le variabili di memoria utilizzate nel programma sono:                **
** CONTAX, CONTAY,                                                       **
** ********************************************************************* **

USE TRASF
  APPEND BLANK
    REPLACE ID WITH "ID",X WITH "PMP"
  APPEND BLANK
    REPLACE ID WITH "B",X WITH "X"
  APPEND FROM VECCHIO
    REPLACE ALL KN WITH 0
    COUNT TO NY
```

```
APPEND FROM VECCHIO


 GOTO NY
 REPLACE ALL KD WITH "*" FOR KN>0
  ** REGISTRAZIONE AREA DI TABELLA                                    **

GOTO 2
  REPLACE  XN WITH 2, Y WITH "Y",YN WITH NY
  STORE 1 TO CONTAX,CONTAY
    GOTO 3
             DO WHILE CONTAX <3
                      DO WHILE CONTAY < NY-1
                         REPLACE ID WITH "C",X WITH "X", XN WITH CONTAX
                         REPLACE Y WITH "Y", YN WITH CONTAY, K WITH 'K'
                         STORE CONTAY + 1 TO CONTAY
                         SKIP
                      ENDDO
                   STORE CONTAX + 1 TO CONTAX
                   STORE 1 TO CONTAY
                ENDDO

  GOTO 2
    STORE CONTAX-1 TO CONTAX
    REPLACE XN WITH CONTAX, YN WITH NY
    APPEND BLANK
         REPLACE ID WITH "E"
    COPY TO TRASFSLK SDF DELIMITED
```

4) Lo stesso programma visto al punto 3 può copiare i dati del file con estensione .DBF in un file con estensione, .TXT in caratteri ASCII leggibili da sistema operativo.

```
. DISPLAY ALL
00001  ID PMP    0       0                        0
00002  B  X      2 Y     8                        0
00003  C  X      1 Y     1 K  EMILIA              0
```

```
00004  C  X      1 Y     2 K  ROMAGNA            0
00005  C  X      1 Y     3 K  VENETO             0
00006  C  X      1 Y     4 K  LOMBARDIA          0
00007  C  X      1 Y     5 K  TOSCANA            0
00008  C  X      1 Y     6 K  PIEMONTE           0
00009  C  X      2 Y     1 K  *            15580900
00010  C  X      2 Y     2 K  *            12850500
00011  C  X      2 Y     3 K  *            15890200
00012  C  X      2 Y     4 K  *            13570900
00013  C  X      2 Y     5 K  *            12300000
00014  C  X      2 Y     6 K  *            14200800
00015  E         0       0                        0
.


  QUIT
*** End run   dBASE II   ***

Remember to back-up your data.


A>TYPE TRASFSLK.TXT
'ID','PMP',   ,' ',   ,' ',' ',        0
'B','X',  2,'Y',  8,' ',' ',        0
'C','X',  1,'Y',  1,'K','EMILIA',        0
'C','X',  1,'Y',  2,'K','ROMAGNA',        0
'C','X',  1,'Y',  3,'K','VENETO',        0
'C','X',  1,'Y',  4,'K','LOMBARDIA',        0
'C','X',  1,'Y',  5,'K','TOSCANA',        0
'C','X',  1,'Y',  6,'K','PIEMONTE',        0
'C','X',  2,'Y',  1,'K','*', 15580900
'C','X',  2,'Y',  2,'K','*', 12850500
'C','X',  2,'Y',  3,'K','*', 15890200
'C','X',  2,'Y',  4,'K','*', 13570900
'C','X',  2,'Y',  5,'K','*', 12300000
'C','X',  2,'Y',  6,'K','*', 14200800
'E',' ',   ,' ',   ,' ',' ',

A>
```

5) Da editore di sistema si debbono sostituire al file che viene qui di seguito presentato le virgole con i punti e virgola ed eliminare alcuni componenti fino ad ottenere il file definitivo.

BREVE RIEPILOGO DEI COMANDI DEL MULTIPLAN

COMANDO SUB-COM.

ALPHA Permette di inserire un testo o un valore numerico nella cella attiva

BLANK Consente di cancellare il contenuto delle celle specificate

COPY RIGHT Permette di definire il numero di celle a destra della cella attuale in cui si vuole copiare il contenuto della cella attiva

DOWN Permette di definire il numero di celle sottostanti la cella attuale in cui si vuole copiare il contenuto della cella attiva

FROM Permette anche la copia di celle non contigue definendo i riferimenti delle celle da copiare nei riferimenti delle celle riceventi

DELETE Consente di cancellare tutta una riga o tutta una colonna del foglio selezionando l'opzione R per riga o C per colonna

EDIT Permette di scrivere o modificare una formula o testi

FORMAT CELLS Permette di definire le coordinate delle celle di cui si vuole modificare il formato di visualizzazione

DEFAULT Permette di definire il formato di visualizzazione e il tipo di allineamento di default delle celle (con il sub-comando cells) o la larghezza di default per tutte le colonne della tabella (con il sub-comando width)

OPTIONS Permette di definire e verificare le opzioni di formato relative alle celle

WIDTH Permette di definire, per colonne specificate, una larghezza diversa da quella di default

GOTO NAME Posiziona il puntatore di cella sulla cella corrispondente al nome specificato

ROW-COL Dopo aver introdotto il numero di riga e di colonna posiziona il puntatore di cella su quella richiesta

WINDOW Posiziona il puntatore di cella su quella di cui è stato specificato il riferimento all'interno della finestra indicata

HELP RESUME Ripristina la tabella attiva dal punto in cui era stato richiamato il comando HELP

START Visualizza la prima pagina del file di HELP

NEXT Visualizza la pagina successiva del file di HELP

PREVIOUS Visualizza la pagina precedente del file di HELP

APPLICATION Visualizza una lista di problemi comuni che possono evidenziarsi nell'utilizzo del Multiplan e i comandi da utilizzare per la loro risoluzione

COMMANDS Visualizza la descrizione di ciascun comando a cominciare dal primo

EDITING Visualizza la descrizione dell'ambiente editing

FORMULAS Evidenzia delle informazioni relative alle regole da seguire e una lista delle funzioni che possono essere utilizzate nella definizione di formule

KEYBOARD Evidenzia una lista di tasti e relativo significato da utilizzare in ambiente Multiplan

INSERT ROW Inserisce nelle posizioni precedenti la riga attiva tutte o parte di una o più colonne con contenuto nullo

COLUMN Inserisce nelle posizioni precedenti la colonna specificata tutte o parte di una o più colonne con contenuto nullo

LOCK CELLS Permette di bloccare/sbloccare una o più celle per prevenire la modifica accidentale dei valori

FORMULAS Permette di bloccare/sbloccare tutte le celle contenenti testi o formule per prevenire che vengano modificate accidentalmente

Y Conferma la richiesta

MOVE ROW Permette di definire i riferimenti relativi a righe da spostare all'interno della tabella e i riferimenti delle posizioni in cui debbono essere spostate

COLUMN Permette di definire i riferimenti relativi a colonne da spostare all'interno della tabella e i riferimenti della posizione in cui debbono essere spostate

NAME Permette di assegnare un nome ad una o più celle secondo quanto specificato nei campi del comando

OPTION Permette di selezionare alcune opzioni che si riflettono sul calcolo della tabella, quali ricalcolo automatico, o la definizione di una cella contenente il "test di completamento" per le operazioni di iterazione

PRINT PRINTER Stampa la tabella attiva secondo il formato definito con i comandi Print Margins e Print Options FILE Memorizza la tabella attiva in un file su disco informato stampabile

MARGINS Permette di definire il formato delle pagine di stampa in termini di margine sinistro, lunghezza di riga di stampa, numeri di righe per pagina

OPTIONS Permette di definire le opzioni di stampa, quali:
stampa solo di una parte della tabella
stampa delle formule anzichè dei valori
soppressione in stampa dei numeri di riga o colonna
definizione di alcuni parametri hardware

QUIT Consente di uscire dalla sezione di Multiplan in corso. Viene richiesta la conferma

Y Per l'esecuzione del comando. La tabella attiva non viene salvata su disco

SORT Permette di definire parte o tutta la colonna i cui valori si vogliono ordinare. L'ordinamento può essere fatto in modo crescente o decrescente, sia per valori numerici che per testi o valori logici

TRANSFERT LOAD Permette di definire un nome di un file che sarà caricato in memoria per l'elaborazione. Utilizzando i tasti di movimento cursore e non specificando il nome viene visualizzata la lista dei files residenti su disco e la selezione del file da richiamare in memoria può essere fatta sulla lista selezionando il file voluto

SAVE Memorizza la tabella attiva su disco attribuendole il nome specificato. Nel caso di unatabella preesistente su disco con lo stesso nome, viene chiesta la conferma per l'esecuzione del comando (Y)

CLEAR Cancella dalla memoria la tabella attiva. Viene chiesta la conferma per l'esecuzione del comando (Y)

DELETE Cancella da disco la tabella specificata. Possono essere utilizzati i tasti di movimento cursore per visualizzare la lista di tutti i files esistenti su disco

OPTIONS Permette di definire l'unità disco di default e il formato di memorizzazione/richiamo delle tabelle

RENAME Permette di salvare la tabella attiva su disco attribuendole un nome diverso. Il comando ridefinisce le eventuali correlazioni esterne

VALUE Introduce la formula

WINDOW SPLIT HORIZONTAL: permette di definire una finestra nell'ambito della tabella. La tabella attiva è splittata orizzontalmente

VERTICAL: permette di definire una finestra nell'ambito della tabella. La tabella attiva è splittata verticalmente

TITLES: permette di definire una finestra contenente i titoli delle righe e colonne visualizzate. La finestra attiva viene suddivisa in 2 o 4 finestre a seconda che i titoli siano relativi solo alle righe o alle colonne o a entrambe

BORDER Consente di riquadrare o eliminare la riquadratura della finestra specificata

CLOSE Elimina la finestra specificata dal video

LINK Permette di definire/eliminare connessioni tra le finestre specificate. La connessione implica lo scrolling simultaneo delle finestre connesse

EXTERNAL COPY Permette di copiare da una tabella un'area delimitata da righe e colonne definita da un nome.

LIST E' riprodotto il nome della tabella che supporta la tabella attiva

USE Introduce il filename. Con CR sostituisce il nome della tabella che viene prodotta

La diffusione del calcolatore nelle aziende di qualunque dimensione, negli uffici dei professionisti e nella scuola, offre nuove opportunità nella gestione della informazione e pone molti nella condizione di dover utilizzare questo strumento per soddisfare importanti esigenze informative.
Infatti, dopo il periodo immediatamente conseguente alla introduzione del calcolatore in azienda, durante il quale la software house o il "centro di calcolo" dichiarano di poter risolvere qualunque problema aziendale, ci si deve scontrare con la realtà delle cose: le personalizzazioni sono costose e difficili, i risultati delle elaborazioni non sempre arrivano in tempo utile e l'utente ha sempre di più l'impressione di dover affrontare i problemi da solo. Tuttavia senza un'adeguata metodologia di approccio alla costruzione dei programmi l'utente è destinato a scontrarsi con altrettante delusioni.
L'autore ritiene di poter intervenire proprio per dare un aiuto a coloro che si chiedono se esistono criteri e tecniche di analisi cui orientarsi nella progettazione dei programmi, fornendo inoltre le routine di uso gestionale più comune in BASIC MS-DOS e Apple Soft nonchè una panoramica sui comandi più frequenti del dBase della ASHON TATE e del MULTIPLAN, con relativi esempi.

L. 27.000 COD 570P ISBN 88-7056-213-1

159

La Contabilità col Personal Computer

GRUPPO EDITORIALE JACKSON